Designers' Guide to Social Simulations, No.1 PlatBox Simulator マニュアル

Chap.1 PlatBox Simulator のしくみ Chap.2 PlatBox Simulator の基本操作 Chap.3 制御パネル Chap.4 World Initializer Chap.5 Sim Iterator Chap.6 DataCollection Manager Chap.7 ReportComponent Manager Chap.8 GraphComponent Manager Chap.9 Log Monitor Chap.10 Profiler Chap.11 Communication Viewer Chap.12 Relation Viewer

PlatBox Project



目次

第1章	PlatBox Simulator のしくみ	1
1.1	PlatBox Simulator の基本構造	1
1.2	PlatBox Simulator のファイル構成	2
1.3	PlatBox プラグインの追加	2
節り音	PlatBox Simulator の基本操作	5
카스무 91	PlatBox Simulator の画面構成	5
2.1	ThatBox Simulator の回面構成	С
2.2		0
	2.2.1 起勤	0
		(
2.3		7
	2.3.1 世界を開く	7
	2.3.2 インポート	8
	2.3.3 エクスポート	8
2.4	シミュレーションの制御	8
2.5	ツールの利用................................	8
	2.5.1 あらかじめ組み込まれている Presentation Component	8
	2.5.2 独自に追加される Presentation Component	9
2.6	ビューアの利用................................	9
	2.6.1 あらかじめ組み込まれている PlatBox プラグイン	9
	2.6.2 独自に追加される PlatBox プラグイン	9
2.7	プロパティの変更・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	9
	2.7.1 速度設定	9
	2.7.2 言語設定	9
2.8	PlatBox Simulator に関する情報の閲覧	10
	2.8.1 PlatBox について	10
	2.8.2 バージョン情報	10
第3章	制御パネル:シミュレーションの実行を制御する	11
3.1	機能と画面構成...............................	11

3.2	シミュ	ュレーションの実行を制御する	12
	3.2.1	シミュレーションを実行する	12
	3.2.2	シミュレーションを一定時間実行する	12
	3.2.3	シミュレーションを停止する	12
	3.2.4	シミュレーションを最初の状態に戻す	12
竺/卉	\		10
	VVORIO	i Initializer:初期値を設定する	13
4.1	機能と	こ 囲 阻 構 成	13
4.2	ハフン	×ーダを設定9る	14
	4.2.1		14
	4.2.2		15
	4.2.3		15
	4.2.4	設定を適用する	15
	4.2.5	設定を削除する・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	15
4.3	設定に	7ァイルを扱う	15
	4.3.1	設定をファイルに保存する........................	15
	4.3.2	PlatBox Simulator を用いずに設定ファイルを用意する	16
	4.3.3	設定をファイルから読み込む	17
第5章	Sim It	terator : シミュレーションを連続実行する	19
第5章 5.1	Sim It 機能と	terator:シミュレーションを連続実行する ≤画面構成....................................	19 19
第5章 5.1 5.2	Sim It 機能と シミニ	terator:シミュレーションを連続実行する ≤画面構成....................................	19 19 21
第5章 5.1 5.2	Sim It 機能と シミニ 5.2.1	terator:シミュレーションを連続実行する ≤画面構成....................................	19 19 21 21
第 5 章 5.1 5.2	Sim It 機能と シミニ 5.2.1 5.2.2	terator:シミュレーションを連続実行する ニ画面構成 ュレーションを連続実行するプ数を設定する 1回のシミュレーションのステップ数を設定する 初期値の変更範囲を設定する	19 19 21 21 21
第 5 章 5.1 5.2	Sim It 機能と シミニ 5.2.1 5.2.2 5.2.3	terator:シミュレーションを連続実行する <回面構成	19 19 21 21 21 21 22
第5章 5.1 5.2	Sim It 機能と シミニ 5.2.1 5.2.2 5.2.3	terator:シミュレーションを連続実行する <回面構成	19 19 21 21 21 22
第 5 章 5.1 5.2 第 6 章	Sim It 機能と シミニ 5.2.1 5.2.2 5.2.3 DataC	terator : シミュレーションを連続実行する < 画面構成	 19 19 21 21 21 21 22 23
第5章 5.1 5.2 第6章 6.1	Sim It 機能と シミニ 5.2.1 5.2.2 5.2.3 Data 機能と	terator : シミュレーションを連続実行する < 画面構成	 19 21 21 21 22 23 23
第5章 5.1 5.2 第6章 6.1 6.2	Sim It 機能と シミニ 5.2.1 5.2.2 5.2.3 Data(機能と Data(terator : シミュレーションを連続実行する < 画面構成	 19 19 21 21 21 21 22 23 23 24
第5章 5.1 5.2 第6章 6.1 6.2 6.3	Sim lt 機能と シミニ 5.2.1 5.2.2 5.2.3 Data 機能と Data Data	terator : シミュレーションを連続実行する < 画面構成	 19 19 21 2
第5章 5.1 5.2 第6章 6.1 6.2 6.3	Sim It 機能と シミニ 5.2.1 5.2.2 5.2.3 Data 機能と Data (0 Data (6.3.1	terator:シミュレーションを連続実行する ユレーションを連続実行する ユロのシミュレーションのステップ数を設定する 初期値の変更範囲を設定する 連続実行して動作を確認する Collection Manager:データを収集する Collection によるデータ収集の流れ Collection の名前を設定する	 19 19 21 21 21 22 23 23 24 26 26
第5章 5.1 5.2 第6章 6.1 6.2 6.3	Sim It 機能と シミニ 5.2.1 5.2.2 5.2.3 Data 機能と Data 6.3.1 6.3.2	terator : シミュレーションを連続実行する ニ 画面構成	 19 19 21 22 23 24 26 26 27
第5章 5.1 5.2 第6章 6.1 6.2 6.3 6.4	Sim It 機能と シミニ 5.2.1 5.2.2 5.2.3 Data (機能と Data (6.3.1 6.3.2 収集し	terator : シミュレーションを連続実行する こ 画面構成	 19 19 21 21 21 21 22 23 23 24 26 26 26 27 31
第5章 5.1 5.2 第6章 6.1 6.2 6.3 6.4 6.5	Sim lt 機能と シミニ 5.2.1 5.2.2 5.2.3 Data (機能と Data) 6.3.1 6.3.2 収集し Data)	terator : シミュレーションを連続実行する 二回面構成	 19 19 21 23 24 26 26 26 26 26 26 27 31 32
第5章 5.1 5.2 第6章 6.1 6.2 6.3 6.4 6.5 6.6	Sim It 機能と シミニ 5.2.1 5.2.2 5.2.3 Data (機能と Data (0 ata) 6.3.1 6.3.2 収集し Data (ブーク)	terator : シミュレーションを連続実行する < 画面構成	 19 19 21 21 21 21 22 23 23 24 26 26 27 31 32 32
第5章 5.1 5.2 第6章 6.1 6.2 6.3 6.4 6.5 6.6	Sim It 機能と シミニ 5.2.1 5.2.2 5.2.3 Data (機能と Data (6.3.1 6.3.2 収集し Data (で一今 6.6.1	terator:シミュレーションを連続実行する コレーションを連続実行する コロのシミュレーションのステップ数を設定する 初期値の変更範囲を設定する 潮崩値の変更範囲を設定する 連続実行して動作を確認する こ Collection Manager:データを収集する こ Collection によるデータ収集の流れ Collection を作成・編集する データの収集方法を設定する たデータを閲覧する ア ア ア ア ア マ ア	 19 19 21 22 23 24 26 27 23 24 26 27 23 24 26 27 23 24 26 26 27 31 24 26 27 31 32 32 32 32
第5章 5.1 5.2 第6章 6.1 6.2 6.3 6.4 6.5 6.6	Sim It 機能と シミニ 5.2.1 5.2.2 5.2.3 Data(機能と Data(6.3.1 6.3.2 収集し Data(データ 6.6.1 6.6.2	terator:シミュレーションを連続実行する ロレーションを連続実行する ロのシミュレーションのステップ数を設定する 初期値の変更範囲を設定する 初期値の変更範囲を設定する 連続実行して動作を確認する こ Collection Manager:データを収集する こ Collection Eよるデータ収集の流れ Collection を作成・編集する アータの収集方法を設定する データを閲覧する アデータを閲覧する アデータを閲覧する アデータ関連の設定をファイルを扱う データ関連の設定をファイルから読み込む	 19 19 21 23 24 26 26 27 24 26 27 27 24 26 27 27 28 29 29 29 29 29 29 24 26 26 2

35

7.1	機能と画面構成・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	35
7.2	ReportComponent を作成する	36
	7.2.1 ReportComponent の名前を設定する	36
	7.2.2 ファイルの出力先と出力名を設定する	37
	7.2.3 データを出力する間隔を設定する	37
	7.2.4 出力するデータを指定する	37
	7.2.5 ReportComponent を追加する	38
7.3	ファイルへの出力を制御する...........................	38
	7.3.1 ReportComponent の動作を開始する	38
	7.3.2 ReportComponent の動作を停止する	39
7.4	ReportComponent を削除する	39
7.5	ReportComponent を保存する	39
7.6	出力されたファイルを見る..........................	39
∽○夽	CraphComponent Managar・データたグラフに出力する	/1
 	GraphComponent Manager. ケータをケノノに山川9 る	+1 /1
0.1		+1 19
8.2	GraphComponent を1Fnx 9 る	+2 19
	0.2.1 石削で設定する · · · · · · · · · · · · · · · · · · ·	±2 19
	0.2.2 山川9のテーツを指定9の ・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	40 49
83	6.2.5 Jーラを山刀りる间隔を設定りる	40 42
0.0	クラブベの山川を削迎する	40 42
	8.3.1 GraphComponent の動作を開始する	±0 4.4
8 1	6.5.2 GraphComponent た脚栓する	±±
0.4 9 5	GraphComponent を招存する	14 15
0.0 9.6	Graph Viewer を使う	40 45
8.0		±0 45
	8.6.2 Craph Viewer のデータ設定を行う	±0 45
	8.6.2 Graph Viewer の f	±0 46
	8.6.4 グラフの X 軸の動きを自動的に追跡する	±0 47
	865 グラフの 戦 な 表示す ろ	17 47
	8.6.6. グラフの全体像を目ろ	17
	8.6.7 グラフを画像ファイルに出力する	±1 47
		*1
第9章	Log Monitor : ログをみる	49
9.1	機能と画面構成・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	49
9.2	出力されるログの項目....................................	49

第 10 章	Profiler:メモリの状態を見る	51
10.1	機能と画面構成・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	51
	10.1.1 Profiler でできること	51
	10.1.2 操作パネル	51
	10.1.3 Instance History 表示パネル \ldots	52
	10.1.4 Memory History 表示パネル	53
10.2	Profiler による観察結果の解釈	53
	10.2.1 オブジェクト数が増加して、空きメモリ量が減少している場合	53
	10.2.2 オブジェクト数は増加していないが、空きメモリ量が減少してい	
	る場合	54
第 11 章	: Communication Viewer : Agent のやりとりを見る	57
11.1	機能と画面構成・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・・	57
11.2	表示を設定する...............................	58
11.3	アニメーションの表示を設定する.......................	58
	11.3.1 全てのアニメーション	59
	11.3.2 個別のアニメーション	59
第 12 章	Relation Viewer : Agent 間の Relation を見る	61
12.1	機能と画面構成...............................	61
12.2	表示項目を設定する	61
	12.2.1 表示する Agent を設定する	61
	12.2.2 表示する Relation を設定する	62
12.3	いろいろな方法で Relation を見る	63
	12.3.1 Agent を円形に並べる	63
	12.3.2 AgentType ごとに一列に並べる	64
	12.3.3 Agent を自由に配置する	65
	12.3.4 Agent を Relation 数の順に並べる	65

第1章

PlatBox Simulator のしくみ

1.1 PlatBox Simulator の基本構造

PlatBox Simulator は、マルチエージェントモデルのシミュレーションを実行するため のプラットフォーム(土台)です。このプラットフォームに、シミュレーションモデルや 表示機能等を組み込むことで、PlatBox Simulator 上で、それらを利用できるようになり ます。このように、モデルや機能のコンポーネント(部品)をプラットフォームに組み込 むことを、「プラグインする」といいます。



図 1.1: PlatBox Simulator の基本構造

PlatBox Simulator にプラグインされるコンポーネントは大きくわけて二種類ありま す。「Model Component」と「Presentation Component」です。Model Component(通称:モデル)は、シミュレーションモデルを部品化したものです。Model Componentは、 ふつう、基礎モデルにもとづいて、Component Builder を用いて作成されます。もう一 方の Presentation Component(通称:プレコン)は、シミュレーションの操作や表示、 および記録を行うための部品です。Presentation Component には、シミュレーションの 状況を文字やグラフィックスで表示するためのものや、マウスやキーボードからシミュ レーションの設定を変更するためのものなどがあります。

これらの Model Component や Presentation Component は、PlatBox Simulator に プラグインするので、「PlatBox プラグイン」と呼ぶことがあります。PlatBox プラグイ ンには、(1)Model Component のみからなる場合、(2)Presentation Component のみか らなる場合、(3)Model Component と Presentation Component を組み合わせてある場 合、の3つの場合があります。



図 1.2: PlatBox プラグインとコンポーネントの関係

1.2 PlatBox Simulator のファイル構成

PlatBox 関連の基本的なファイルを説明しておくことにしましょう。それらのファイル は、インストール時に設定を変えていなければ、C:\Program Files\PlatBoxSimulator にあります。

まず、「PlatBoxSimulator.exe」が、PlatBox Simulatorの起動に用いる実行ファイルで す。これをダブルクリックすることで、PlatBox Simulatorの本体である「platbox.jar」 が実行されます。「plugins」フォルダは、PlatBox プラグインを入れるためのフォルダ です。

1.3 PlatBox プラグインの追加

PlatBox Simulator に PlatBox プラグインを追加すると、Model Component の World を読み込んだり、Presentation Component を用いてシミュレーションの制御、結果の表 示などができるようになります。

PlatBox プラグインは、ファイルとしては、JAR (JavaARchive) という圧縮ファイルに なっています(ファイル名は「~.jar」です)。この JAR ファイルを、PlatBox Simulator

🚞 PlatBoxSimulator		
ファイル(E) 編集(E) 表示(V)	お気に入り(<u>A</u>) ッ・	ール(① ヘルプ(巴) 🛛 🥂
🔆 F3 • 🕗 • 🏂 🔎	検索 🍺 フォルダ	🕞 🍞 🗙 🦷 👋
アドレス(D) 🗀 C:¥Program Files¥	PlatBoxSimulator	💌 芛 移動
名前	サイズ	種類 ▲
jvm		ファイル フォルダ
uninst		ファイル フォルダ
🛅 api		ファイル フォルダ
Conf		ファイル フォルダ
🛅 lib		ファイル フォルダ
🛅 plugins		ファイル フォルダ
🖬 platbox.jar	969 KB	JAR ファイル
🖬 .properties	8 KB	PROPERTIES ファイル
🖀 box.ico	5 KB	アイコン
PlatBoxSimulator.exe	329 KB	アプリケーション
📄 readme.txt	4 KB	テキスト ドキュメント
🚺 platboxSrc.zip	393 KB	圧縮 (zip 形式) フォルダ
•		Þ

図 1.3: PlatBox のファイル構成

をインストールしたディレクトリ(通常は C:\Program Files\PlatBoxSimulator)に 存在する plugins フォルダにコピー(移動)することで、プラグインできます。プラグイ ンしたものは、次回の PlatBox Simulator 起動時から実行することができます。



図 1.4: PlatBox プラグインをプラグインする

第2章

PlatBox Simulator の基本操作

2.1 PlatBox Simulator の画面構成

PlatBox Simulator を起動すると、「制御パネル」(第3章: p.11)が開かれた状態で起動 します。画面の上部には、「メニューバー」があります。メニューバーには、「ファイル」、 「制御」、「ツール」、「ビューア」、「プロパティ」、「ヘルプ」の6つのメニューがあります。



図 2.1: 起動時の PlatBox Simulator

各メニューに含まれる項目の一覧は、以下の通りです。

「ファイル」メニュー

- [インポート] シミュレーションの実行途中の状態を読み込みます。
- [エクスポート] シミュレーションの実行途中の状態を保存します。
- [終了] PlatBox Simulator を終了します。

「制御」メニュー

- [実行] 停止中のシミュレーションを実行します。
- [一定時間実行] シミュレーションを実行し、「実行設定」で設定した時間 が経過したら自動的に停止します。
- [停止] 実行中のシミュレーションを停止します。
- [リセット] シミュレーションを初期状態 (World を読み込んだときの状態) に戻します。パラメータの設定もデフォルトの設定(第4章: p.13)に戻ります。
- [実行設定] 「一定時間実行」で実行する時間を設定します。
- 「ツール」メニュー

特殊な操作等のツールが利用できます。詳しくは、第2.5節: p.8を参照してください。

「ビューア」メニュー

経過表示のビューアが利用できます。詳しくは、第2.6節: p.9を参照してください。

「プロパティ」メニュー

[速度設定] シミュレーションの実行速度を設定します。

- [言語設定] PlatBox Simulator で使用する言語を設定します。
- 「ヘルプ」メニュー

[PlatBox について] PlatBox Project について紹介します。

[バージョン情報] PlatBox Simulator のバージョン情報を表示します。

2.2 PlatBox Simulator の起動と終了

2.2.1 起動

PlatBox Simulator を起動するには、「スタート」メニューから「すべてのプログラ ム」を選択して「PlatBox Simulator」をクリックするか、デスクトップにある「PlatBox Simulator へのショートカット」をダブルクリックします。PlatBox Simulator が起動す ると、図 2.1 のようなウィンドウが開きます。 2.2.2 終了

PlatBox Simulator を終了するには、メニューバーにある「ファイル」から「終了」を 選択するか、ウィンドウ上部の「閉じる」ボタン(×)をクリックします。

PlatBox Simulator を終了させると、それまで読み込まれていたシミュレーションの状態は破棄されるので、シミュレーションの状態を保存したい場合はエクスポートを行って ください(第2.3.3節: p.8)

2.3 ファイルの読み込みと保存

2.3.1 世界を開く

PlatBox Simulator に、モデルの World を読み込みます。メニューバーにある「ファ イル」から「世界を開く」を選択すると、「世界の選択」ダイアログが開きます。その中 から、PlatBox Simulator で読み込む World を選択します。フォルダは、そのフォルダ 名の部分をダブルクリックするか、そのすぐ横にある「+」をクリックすることで開くこ とができます。World を選択したら、「了解」ボタンを選択します(図 2.2)。

世界の選択	×
E-monorg	
artificialmarket	
BakerWorld	

図 2.2: 「世界の選択」ダイアログ

World を開く際の注意	
新しい World を読み込むと、それまで読み込まれていた World は破棄されます。	

2.3.2 インポート

PlatBox Simulator に、Box ファイル (拡張子が「.box」のファイル)を読み込みます。 保存した時点の状態から、シミュレーションを再開することができます。

メニューバーにある「ファイル」から「インポート」を選択すると、「開く」ダイアロ グが開きます。Box ファイルがあるフォルダに移動し、ファイルを選択して、「開く」を 押します。シミュレーションの進行していた時間や、パラメータの値、World に存在する Agent の状況などが、ファイルから読み込まれます。

2.3.3 エクスポート

シミュレーションの途中の状態を、Box ファイル (拡張子が「.box」のファイル) に保 存します。

メニューバーにある「ファイル」から「エクスポート」を選択すると、「保存」ダイアロ グが開きます。保存先とファイル名を設定し、「保存」を押します。シミュレーションの 進行した時間や、パラメータの値、World に存在する Agent の状況などがファイルに書 き出されます。

2.4 シミュレーションの制御

制御の操作については、第3章: p.11を参照してください。

2.5 ツールの利用

2.5.1 あらかじめ組み込まれている Presentation Component

PlatBox Simulator には、次に挙げる Presentation Component が、「ツール」として 標準搭載されています。これらは、World の操作やデバッグなどのためのツール群です各 ツールの使い方については、それぞれの解説章を参照してください。

- 制御パネル(第3章: p.11)
- World Initializer (第4章: p.13)
- Sim Iterator (第5章: p.19)
- DataCollection Manager (第6章: p.23)
- ReportComponent Manager (第7章: p.35)
- GraphComponent Manager (第 8 章: p.41)
- Log Monitor (第9章: p.49)
- Profiler (第 10 章: p.51)

2.5.2 独自に追加される Presentation Component

読み込む World によっては、独自の Presentation Component が「ツール」に追加される場合があります。

2.6 ビューアの利用

2.6.1 あらかじめ組み込まれている PlatBox プラグイン

PlatBox Simulator には、次に挙げる Presentation Component が、「ツール」として 標準搭載されています。これらは、シミュレーションの結果を見て分析するためのビュー アです^{*1}。各ビューアの使い方については、それぞれの解説章を参照してください。

- Communication Viewer (第11章: p.57)
- Relation Viewer (第12章: p.61)

2.6.2 独自に追加される PlatBox プラグイン

読み込む World によっては、独自の Presentation Component が「ビューア」に追加 される場合があります。

2.7 プロパティの変更

2.7.1 速度設定

メニューバーにある「プロパティ」から「速度設定」を選択すると、シミュレーション の実行速度を設定できます。シミュレーションの速度は、「速い」、「普通」、「遅い」の3 つから選択します。

速度設定を「速い」にする場合の注意
 「速い」に設定すると CPU に負荷が掛かるので、PC が行っている他の処理速度に影響
 を与えます。状況に応じて使い分けるとよいでしょう。

2.7.2 言語設定

メニューバーにある「プロパティ」から「言語設定」を選択すると、PlatBox Simulator で使用する言語を設定できます。「デフォルト (OS の設定を利用)」、「日本語」、「英語」の

^{*&}lt;sup>1</sup> グラフは、シミュレーションの結果を見るものですが、グラフ出力の設定を行う GraphComponent Manager から呼び出して開きます。GraphComponent Manager は、グラフへの出力を管理するツー ルなので、「ビューア」ではなく「ツール」に含まれます。

3つから選択します。その設定は、次回 PlatBox Simulator を起動したときから有効になります。

2.8 PlatBox Simulator に関する情報の閲覧

2.8.1 PlatBox について

メニューバーにある「ヘルプ」から「PlatBox について」を選択すると、PlatBox Project の紹介や URL などの情報が表示されます。

2.8.2 バージョン情報

メニューバーにある「ヘルプ」から「バージョン情報」を選択すると、バージョン情報 が表示されます。

第3章

制御パネル:シミュレーションの実 行を制御する

3.1 機能と画面構成

制御パネルは、シミュレーションの実行を制御するためのプラグインです。シミュレー ションの実行、停止、リセットを行うことができます。また、実行設定を行うことで、設 定した一定の時間だけ実行することも可能です。

制御パネルは、図 3.1 のような画面構成になっています。それぞれの部分の機能は以下 の通りです。



図 3.1: 制御パネル

①World 表示領域	読み込まれている World の名前を表示します。 Wolrd が読み込まれていない場合は「世界が読み
	込まれていません」と表示されます。
②「実行」ボタン	停止中のシミュレーションを実行します。
③「一定時間実行」ボタン	シミュレーションを「実行設定」で設定した時間
	だけ実行して停止します。
④「停止」ボタン	実行中のシミュレーションを停止します。
⑤「リセット」ボタン	シミュレーションの時間を最初に戻します。パラ
	メータの設定もデフォルトの設定(第4章: p.13)
	に戻ります。
⑥「時間表示」ボタン	現在の時間と「実行設定」ボタンを表示します。
⑦「実行設定」ボタン	「一定時間実行」で実行する時間を設定するダイ
	アログを表示します。

- 3.2 シミュレーションの実行を制御する
- 3.2.1 シミュレーションを実行する

停止中のシミュレーションを実行するには、「実行」ボタンを押します。

3.2.2 シミュレーションを一定時間実行する

シミュレーションを一定の時間だけ実行して停止するには、「一定時間実行」ボタンを 押します。どれだけ実行するのかの設定は、「実行設定」ボタンで開く「実行設定」ダイ アログで行います。

3.2.3 シミュレーションを停止する

実行中のシミュレーションを停止するには、「停止」ボタンを選択します。「停止」して いる状態で「実行」すると、停止した時間からシミュレーションが再開します。

3.2.4 シミュレーションを最初の状態に戻す

シミュレーションを最初の状態に戻すには、「リセット」ボタンを選択します。進行していた時間、パラメータの初期値、World に存在する Agent の状態などはすべて最初の状態に戻ります。

🐔 シミュレーションを最初の状態に戻す際の注意

制御パネルの「リセット」ボタンを選択するなど、シミュレーションの時間が最初に戻 る処理を行う場合、時間だけではなくパラメータの初期値なども最初の状態 (World を 読み込んだ時点での状態) に戻ります。World Initializer (第4章: p.13)で新しい設 定を適用した場合もデフォルトの設定 (第4.2.1章: p.14)に戻るので、注意してくだ さい。

第4章

World Initializer:初期値を設定する

4.1 機能と画面構成

World Initializer は、シミュレーションの初期値を設定するためのプラグインです。 World Initializer では、World Composer で設定した World パラメータの初期値を変更 することができます。World Initializer は、図 4.1 のような画面構成になっています。



図 4.1: World Initializer の画面構成

それぞれの部分の機能は以下の通りです。

①「設定ファイル」メニュー	「開く」と「保存」が選択できます。設定を外
	部ファイルから読み込んだり、外部ファイル
	へ書き込みを行い(第4.3節: p.15)
② 設定リスト	選択可能な設定の名前が表示されます。
③「設定を追加」ボタン	設定リストで選択している設定をコピーして、
	新しい設定を作成します(第4.2.2節:p.15)。
④「設定を削除」ボタン	設定リストで選択している設定を削除します
	(第4.2.5節:p.15)。
⑤ 設定名入力欄	設定の名前を入力します。
⑥ パラメータ・リスト	設定リストで選択している設定に含まれるパ
	ラメータとその値が表示されます。
⑦ 「この設定を適用」 ボタン	パラメータ・リストに表示されているパラメー
	タの初期値を適用します(第 4.2.4 節: p.15)。

PlatBox Simulator に World を読み込むと、その World のパラメータが「デフォルトの設定」として World Initializer に読み込まれます。設定を変えてシミュレーションを行う場合は、すでにある設定(例えば、「デフォルトの設定」)をコピーして作成します。

World Initializer の設定は、World Setting ファイル (拡張子が「.ws」のファイル) と して保存することができます。



4.2 パラメータを設定する

4.2.1 デフォルトの設定を使う

PlatBox Simulator で World を読み込んだときのパラメータの初期値のことを、「デフォルトの設定」と呼びます。World Initializer を開くと、デフォルトの設定を確認する ことができます。制御パネルの「リセット」ボタンを選択したときなど、シミュレーションを最初の状態に戻した場合、デフォルトの設定が適用されます。

World Initializer で設定を編集するには? デフォルトの設定は、編集することはできません。そのため、デフォルトの設定の入力 欄は灰色で表示されています。新しく設定を行いたい場合には、必ずデフォルトの設定 をコピーしてから行ってください。

4.2.2 設定をコピーする

新しい設定を作成するには、すでに設定リストにある設定をコピーしてから行います。 コピーしたい設定を設定リストから選択し、「設定を追加」ボタンを押すと、設定リスト にコピーが追加されます。

コピーした設定には、一時的に「~のコピー」という名前が付きますが、どのような設 定なのか分からなくなるので、分かりやすい名前を付けましょう。設定名入力欄に新しい 名前を入力した後、別の場所を選択すると設定リストも更新されます。

4.2.3 初期値を設定する

それぞれの設定で各パラメータの初期値を変更するには、値の入力欄に新しい初期値を 入力します。新しい値は、入力欄の上に表示されている種類の値である必要があります。 誤った種類の値を入れたり、空欄となった入力欄は赤色で表示されます。すべての入力欄 が正しい種類の値で入力されていることを確認してください。

4.2.4 設定を適用する

変更した設定をシミュレーションの初期値として用いるには、設定リストから設定を選 択し、「この設定を適用」ボタンを押します。「世界を初期化します。よろしいですか?」 と聞かれるので、「はい」を選択します。シミュレーションの時間が最初に戻り、その初 期値が適用されます。

適用した後に、制御パネルなどでリセットした場合には、ここで設定した初期値ではな く、Worldを読み込んだときの「デフォルトの設定」に戻るので注意してください。

4.2.5 設定を削除する

設定リストから、削除したい設定を選択し、「設定を削除」ボタンを選択すると、その 設定が削除されます。現在適用している設定を削除しても、適用した設定は取り消されま せん。

4.3 設定ファイルを扱う

4.3.1 設定をファイルに保存する

World Initializer は、作成したすべての設定をファイルに保存することができます。設 定は、パラメータと値がカンマで区切られている World Setting File 形式(拡張子.ws) で保存されます。

World Initializer ウィンドウの「設定ファイル」メニューから「保存」を選択すると、

「保存」ダイアログが表示されます。保存場所とファイル名を設定して、「保存」ボタンを 選択します。

この保存作業を行わないと設定は保存されません。ただし、PlatBox Simulator で同じ World を扱っている間だけ、設定が保持されています。

4.3.2 PlatBox Simulator を用いずに設定ファイルを用意する

設定ファイルは、World Initializerの保存機能を用いなくても、表計算ソフト (Microsoft Excel など) やテキストエディタを用いて作成・編集することができます。World Setting File 形式は、次の形式で記述されています。

- 1. 1行目にはパラメータの名前をカンマ区切りで入力します。ただし、一つ目の要素 は SettingName などの分かりやすい名前を入力します。1行目には、設定を読み 込みたい World のパラメータがすべて正確に含まれていなければなりません。
- 2.2行目以降は、一行を一つの設定として値をカンマ区切りで入力します(ただし、 1行目で入力したパラメータの順番に合わせます)。一つ目の要素は設定の名前を 入力します。

たとえば、次のようなパラメータと、作りたい設定があった場合には、以下のような内 容のファイルを作成すればよいということになります。

	設定1	設定 2	設定 3
defaultRandomSeed	0	0	1
timeEventRandomSeed	2	1	1

この例の場合、パラメータには defaultRandomSeed と timeEventRandomSeed があ るので、1 行目は、次のように書きます。

Setting Name, default Random Seed, time Event Random Seed

2行目以降は、設定1から設定3までの値を、1行目で書いた順番に対応させて入力します。

SettingNa	ame, defaultRandomSeed, timeEventRandomSeed
設定1,0,2	2
設定2,0,1	
設定 3 ,1,1	l

World Initializer の保存機能で保存したファイルを編集する場合も、以上の記法と同様です。入力が終わったら、ファイルを保存してください。拡張子を「.ws」としておくと、 読み込むときに分かりやすくて便利です。 表計算ソフトで World Setting File 形式のファイル編集する際の注意 表計算ソフトを用いて World Setting File 形式のファイルを編集する際には、保存する 場合も読み込む場合も、カンマ区切りのファイル形式(拡張子.csv)として扱ってくだ さい。表計算ソフト依存のファイル形式(拡張子.xls など)で保存しないように注意が 必要です。

4.3.3 設定をファイルから読み込む

World Initializer では、World Setting File 形式の記法(第 4.3.2 節: p.16)のファイ ルから設定を読み込み、設定リストに追加することができます。

World Initializer ウィンドウの「設定ファイル」メニューから「開く」を選択すると、 「開く」ダイアログが表示されます。そして、ファイルの場所とファイル名を設定して、 「開く」ボタンを選択します。ファイルの拡張子が.ws ではない場合は、「開く」ダイアロ グの「ファイルタイプ」欄から「すべてのファイル」を選択して、一覧表示させてから選 択します。

ファイルから読み込まれた設定は、World Initializerの設定リストの一番下に追加されます。

第5章

Sim Iterator:シミュレーションを連 続実行する

5.1 機能と画面構成

Sim Iterator は、シミュレーションを連続して実行するためのプラグインです。Sim Iterator では、パラメータの初期値を少しずつ変えて、シミュレーションを実行していく ことができます。ReportComponent Manager(第7章: p.35)と組み合わせて用いるこ とで、シミュレーションを連続実行しながら、その結果を外部ファイルに出力することが できます。Sim Iterator は、図 5.1 のような画面構成になっています。

	Dim Iterator	
	Worldの名前:ArtificialMarket ArtificialMarket	
1-	Working to BR	
2-	₹ 7: 100	
	✓ defaultRandomSeed	
	defauttRandomSeed (ring) timeEventRandomSeed	スケール: 1
3-		
	randomTraderNum	
	trandTraderNum initialTraderMonev	
	InitialTraderStock	
	defaultRandomSeed	
	defaultRandomSeed	
	4	

図 5.1: Sim Iterator の画面構成

それぞれの部分の機能は以下の通りです。

1	World の説明	World の名前と World の説明が表示されま
		す。
2	ステップ数入力欄	連続してシミュレーションを実行する際に、
		一回のシミュレーションを何ステップ実行す
		るかを設定します(第 5.2.1 節: p.21)。
3	初期値の変更範囲設定欄	どのパラメータを、どの範囲で変えるかを設定
		します。設定した範囲の中で、一回のシミュ
		レーションごとに変える値の幅(スケール)も
		設定します(第 5.2.2 節: p.21)。
4	「 OK 」 ボタン	連続実行の状況を表示し、連続実行を制御す
		るする Dynamic Consecutive Execute ウィ
		ンドウを表示します。

Dynamic Consecutive Execute ウィンドウは、図 5.2 のような画面構成になっています。

Dynamic Consecutive Executer 連続実行開始 Worldを実行中・・・ history=0 1/5 <20%> 終了 (残り0'30) Worldを実行中・・・ history=2 2/5 <40%> 終了 (残り0'30) Worldを実行中・・・ history=4	
連続実行開始 Worldを実行中・・・ history=0 1/5 <20%> 終了 (残り0'30) Worldを実行中・・・ history=2 2/5 <40%> 終了 (残り0'30) Worldを実行中・・・ history=4	
Worldを実行中・・・ history=0 1/5 <20%> 終了 (残り0'30) Worldを実行中・・・ history=2 2/5 <40%> 終了 (残り0'30) Worldを実行中・・・ history=4	
Worldを実行中・・・ history=0 1/5 <20%> 終了 (残り0'30) Worldを実行中・・・ history=2 2/5 <40%> 終了 (残り0'30) Worldを実行中・・・ history=4	
nistory=0 1/5 <20%> 終了 (残り0'30) Worldを実行中・・・ history=2 2/5 <40%> 終了 (残り0'30) Worldを実行中・・・ history=4	
Worldを実行中・・・ history=2 2/5 <40%> 終了 (残り0'30) Worldを実行中・・・ history=4	
Worldを実行中・・・ history=2 2/5 <40%> 終了 (残り0'30) Worldを実行中・・・ history=4	
history=2 2/5 <40%> 終了 (残り0'30) Worldを実行中・・・ history=4	
2/5 <40%> 終了 (残り0'30) Worldを実行中・・・ history=4	
Worldを実行中・・・ history=4	
history=4	
	(1
-	

図 5.2: Dynamic Consecutive Execute ウィンドウ

それぞれの部分の機能は以下の通りです。

1) 状況表示欄 連続実行の状況を表示します。
 2) 制御ボタン 連続実行中には「停止」ボタン、停止中には「実行!」ボタンが表示され、連続実行を制御します。

5.2 シミュレーションを連続実行する

5.2.1 1回のシミュレーションのステップ数を設定する

Sim Iterator で連続実行する場合、一回のシミュレーションをどれだけの時間行うかを 設定する必要があります。この実行時間は、ステップ数入力欄に入力します。ここで設定 した時間だけ、シミュレーションが実行されると、時間が最初に戻り、初期値を変えて次 のシミュレーションが開始されます。

Sim Iterator の時間設定に関連する注意 ここで行った設定は、制御パネルの実行設定(第3.1節: p.11)を上書きするので注意 してください。

5.2.2 初期値の変更範囲を設定する

Sim Iterator で連続実行する場合、どのパラメータを、どの範囲で変更するのかを設定 する必要があります。設定は、初期値の変更範囲設定欄で行います。まず、どのパラメー タの初期値を変更するかを決めます。初期値の変更範囲設定欄にあるプルダウンメニュー から、初期値を変更するパラメータを選択します。パラメータを選択すると、初期値の変 更範囲設定欄の左側に、値の種類が表示されます。

次に、初期値を変更する範囲を決定します。変更範囲の初期値(以上)入力欄と限界値 (未満)入力欄に、正しい種類の値を入力します。次のシミュレーションに移るときに、ス ケール入力欄に入力した値が初期値に加算され、新たな初期値となります。設定が正しく ない場合、選択したパラメータ名の左側に×印が表示されます。

二つ以上のパラメータの初期値を変更する場合にも、同じ手順で設定を行います。パラ メータは四つまで選択可能です。

Sim Iterator で変更するパラメータを設定する際の注意 設定が正しくない場合や、同じパラメータを複数選択している場合には、実行すること ができません。

各回のシミュレーションでは、パラメータのうち、一つの値だけが変更されます。Sim Iterator がどのように動作するのかを理解するために、ここでは簡単な例で考えてみるこ とにしましょう。パラメータ A を 0 から 2 の範囲(スケールは 1)で、パラメータ B を 10 から 12 の範囲(スケールは 1)で連続実行した場合、各回の初期値は次のようになり ます。

回数	А	В
1回目	0	10
2回目	0	11
3回目	1	10
4回目	1	11

5.2.3 連続実行して動作を確認する

パラメータの初期値設定欄で正しく設定を行い、Sim Iterator の「OK」ボタンを押す と、Dynamic Consecutive Execute ウィンドウが表示されます。Dynamic Consecutive Execute ウィンドウの「実行!」ボタンを選択すると連続実行が開始されます。連続実行を 開始すると、Dynamic Consecutive Execute ウィンドウ右上の終了アイコンが表示され なくなります。

状況表示欄には最初何も表示されていませんが、連続実行を開始すると「World を実行 中・・・」と表示されます。その下に、変更するように設定したパラメータと、その回の 初期値が表示されます。変更するすべてのパラメータが表示された下に、進捗状況と残り 時間の目安が表示されます。

連続実行中に Dynamic Consecutive Execute ウィンドウの「停止」ボタンを選択する と、連続実行が止まります。再び「実行!」ボタンを選択すると 最初の回から 実行され ます。

すべての回が終了すると、Dynamic Consecutive Execute ウィンドウの状況表示欄に 「連続実行終了」と表示されます。

Sim Iterator で連続実行中にエラーが起きたときには、普通の実行時と同様にシミュ レーションが停止します。連続実行する場合、CPU への負荷が大きくなり、不具合が起 きる場合があります。負荷が大きい場合は、PlatBox Simulator のプロパティで速度設定 を「普通」か「遅い」に設定すると改善されることがあります(第 2.7.1 節: p.9)。

第6章

DataCollection Manager:データを 収集する

6.1 機能と画面構成

DataCollection Manager は、シミュレーション実行時に収集されたデータの集合 「DataCollection」を管理するためのプラグインです。DataCollection Manager では、 DataCollection の作成や削除を行うことができます。また、DataCollection が収集した データの表示や編集を行うことができます。

ここで作成・編集した DataCollection は、外部ファイルに保存することができます (Data Collection Setting 形式、拡張子 .dc)^{*1}。また、Data Collection Setting 形式で作 成したファイルから、DataCollection を読み込むことも可能です。



DataCollection Manager は、図 6.1 のような画面構成になっています。

図 6.1: DataCollection Manager の画面構成

^{*1} **E** ここで保存されるのは DataCollection だけでなく、ReportComponent (第7章: p.35) や GraphComponent (第8章: p.41) も一緒に保存されます。

各部分の機能は、以下の通りです。

① DataCollection リスト	作成した DataCollection の一覧が表示されて
② 「 作成 」 ボタン	います。 「DataCollection の作成」ウィンドウを表示し、 DataCollection を作成することができます(第
③「 編集」ボタン	6.3 節: p.26)。 「DataCollection の作成」ウィンドウを表示し、
	DataCollection を編集することができます(第 6.3節: p.26)
④ 「閲覧」ボタン	選択している DataCollection が収集したデー タの一覧を表示します(第 6.4 節: p.31)。
⑤「削除」ボタン	選択している DataCollection を削除します(第 6.5節: p.32)。
⑥ 「 保存 」 ボタン	「Data Collector Setting Save Wizard」ダイ アログを表示し、DataCollection などのデータ 関連の設定をファイルに保存することができま す(第661節: p.22)
⑦ 「読込」ボタン	9 (

6.2 DataCollection によるデータ収集の流れ

DataCollection では、指定した AgentType の Agent 数や、条件を満たす Agent 数、 Agent のもっている情報が含む値など、様々なものをデータとして収集できます。Data-Collection には、すべて「Collector タイプ」を設定する必要があります。Collector タイ プには、次のようなものがあります (これらの Collector タイプの詳しい使い方について は、第 6.3.2 節: p.27 を参照してください)。

Collector タイプ	説明
すべてのエージェント	すべての Agent を収集します。他の DataCollection
	を必要としません。
オブジェクト操作	オブジェクトの属性を収集したり、PlatBox 基礎モデ
	ルに定義されたメソッドによって取得する値を収集
	することができます。データを集めるのに必要なオブ
	ジェクトを含む DataCollection が必要です。
取引	取引された Goods の量を、取引ごとに収集します。他
	の DataCollection を必要としません。
条件抽出	DataCollection の指定した列と条件を照合して、条件
	に合う行のみを抽出します。条件と照合するための列
	を含む DataCollection が必要です。
計算	DataCollection が含む値の四則演算を行います。数値
	を含む DataCollection が必要です。二つの DataCol-
	lection から列を指定することができます。
集計	DataCollection の指定した列のすべての行を集計しま
	す。集計のための列を含む DataCollection が必要で
	す。

DataCollection では、基本的には、別の DataCollection で収集したデータを元にデー タを収集します。作成する DataCollection が、どの Collector タイプであるかによって、 必要となる元データの DataCollection の Collector タイプは異なります。Collector タイ プ間の利用関係は、図 6.2 のようになります。矢印が指している Collector タイプを用い るためには、矢印の元となっているいずれかの Collector タイプの DataCollection が必 要になります。



図 6.2: DataCollection Collector タイプの利用関係

例えば、すべての Agent がそれぞれどの AgentType であるかを収集する場合は、まず 最初に、すべての Agent を収集する DataCollection(「すべてのエージェント」タイプ) を作成します。次に、その DataCollection の一列目 (Agent が一行ごとに入っています) の Type 属性を収集する新たな DataCollection(「オブジェクト操作」タイプ) を作成し ます。その後、ある特定の AgentType の Agent だけを抽出して、その Agent がもって いる Information を収集し、その Information の値を調べるといったことが可能です。

収集したデータは、DataCollection Managerの閲覧機能によって、テーブル表示でき ます。DataCollection を利用して、別の DataCollection を作成する場合には、データ元 となる DataCollection を閲覧して、意図したデータが収集できているかどうかを確認し てから作成することにしましょう (閲覧の方法は、第 6.4 節: p.31 参照を参照してくだ さい)。

DataCollection に収集したデータを、ファイルやグラフに出力するには DataCollection が収集したデータは、ReportComponent Manager を用いてファイル への出力することができます。また、GraphComponent Manager を用いてグラフへ出 力することもできます。ReportComponent Manager については、第7章: p.35 を、 GraphComponent Manager については、第8章: p.41 を参照してください。

6.3 DataCollection を作成・編集する

6.3.1 DataCollection の名前を設定する

DataCollection の作成は、「DataCollection の作成」ウィンドウ(図 6.3)で行います。 DataCollection Manager の「作成」ボタンを選択するか、DataCollection を選択して 「編集」ボタンを押して表示させます。

D	ataCollectionの作成		×
		DataCollectionの作成	
	名前:	New Data	
	Collectorタイプ:	すべてのエージェント <mark>すべてのエージェント</mark> オブジェクト操作 - 属性 オブジェクト操作 - 操作 2011	
		¥251 条件抽出 計算 - 難算 計算 - 加算 計算 - 減算	
		了解取消し	

図 6.3: 「DataCollection の作成」ウィンドウ

まず最初に、DataCollection がどのようなデータを収集しているのかを考え、分かり

やすい名前を付けましょう。「DataCollectionの作成」ウィンドウの「名前」欄に任意の 名前を付けます。

DataCollectionの名前は、どのように付ければよいか?
 DataCollectionの名前の付け方には決まりはありません。ただし、
 DataCollectionには、分かりやすい名前を付けておかないと、後で何のデータかがわからなうなり、混乱を生みます。
 そこで、DataCollectionには「取引」とうような抽象的な名前ではなく、「Moneyと Breadの取引」のように、具体的な名前を付けましょう。また、ファイルから
 DataCollectionを読み込む場合には、同じ名前のDataCollectionがあると不都合が生じる場合があるので、注意してください。

なお、DataCollectionの名前を変更した場合、そのDataCollectionを元につくられた DataCollectionの参照先の名前は、自動的に変更されます。

6.3.2 データの収集方法を設定する

DataCollection がどのようなデータを収集するのかについては、Collector タイプに よって指定します。Collector タイプには、大きく分けて「すべてのエージェント」、「オ プジェクト操作」、「取引」、「条件抽出」、「計算」、「集計」の6種類があります。

Collector タイプの種類によっては、「DataCollection の作成」ウィンドウに新たな項目 が表示されます。すべての項目を適切に設定し、「了解」ボタンを押すことで、DataCollection を作成できます。Collector タイプによっては、すでにある DataCollection を元 に作成するものもあります。どの DataCollection を用いるのかは、「ソース」欄で指定し ます。

「すべてのエージェント」タイプ

「すべてのエージェント」タイプは、すべての Agent を収集します。閲覧時には、一列 目の各行に Agent を表示します^{*2}。このタイプは、他の DataCollection を必要としませ ん。なお、「オブジェクト操作」タイプを用いるためには、この「すべてのエージェント」 タイプの DataCollection が必要となります。

「オブジェクト操作」タイプ

「オブジェクト操作」タイプには、オブジェクトの属性*3を取得する「属性」と、PlatBox 基礎モデルのメソッドを利用してデータを取得する「操作」の2種類があります。このタ

^{*&}lt;sup>2</sup> Agent を表示すると [org.platbox.simulator.model.fmfw.Agent@1ed13da] のような ObjectID が 表示されます(@の後は Agent によって異なります)。 ObjectID が表示されたものは、「オブジェクト 操作 - 属性」 Collector タイプで属性を取得することができます。

^{*3} オブジェクトがもつ値のこと。詳しくはオブジェクト指向に関する文献を参照してください。

イプの DataCollection を作成するにはオブジェクトを含む DataCollection が必要です。 「属性」と「操作」のどちらの場合も、データを取得するオブジェクトを指定します。オ ブジェクトを指定するには、「ソース」欄でオブジェクトを含む DataCollection を指定し、 「対象列」欄でオブジェクトを含む列を指定します。次に、どの「属性」または「操作」を 収集するのかを指定します。「オブジェクト操作」タイプの DataCollection は、「ソース」 欄に指定した DataCollection に1列加えたデータになります。行の増減はありません。

(「オブジェクト操作 - 属性」の場合)

「属性」欄に取得する属性を入力します。「属性リスト」欄から属性を選択した場合には、 「属性」欄は自動的に入力されます。対象となるオブジェクトがどのような属性を持って いるか分からず、しかも「属性リスト」欄には表示されない場合は、対象となるオブジェ クトを含む DataCollection を閲覧して調べることができます。調べるには、まず、表示 されるデータの一覧から、対象となるオブジェクトをダブルクリックして、新しいウィン ドウを開きます。そのウィンドウ上で、「Key」列に示されるのがオブジェクトの属性で、 「Value」列に示されるのが各属性の値です。

(「オブジェクト操作 - 操作」の場合)

「操作」の場合には、「操作」欄からメソッドを選択します。「操作」欄には、対象となるオ ブジェクトに応じた PlatBox 基礎モデルのメソッドが表示されています。ここで主に使 うのは、「get」で始まるメソッドです。例えば、Agent オブジェクトで getAllRelations() を選択すると、その Agent がもつすべての Relation が収集できます。

メソッド名の後の括弧の中に、引数がないメソッドの場合には(例えば、getAllRelations()など)、「操作」欄の下に[引数なし]と表示されます。括弧の中に引数があるメ ソッドの場合には(例えば、getRelations(RelationType)など)、「操作」欄の下に引数を 設定するためのボタンが表示されます。

引数は1つ目から順に「引数 [0]」、「引数 [1]」のように表示されます。値が設定されて いない引数は「null」が示されているので適切な値を設定します。値(null)の横に表示 されている「…」のボタンを押すと、引数を選択するダイアログが表示されます。以降、 画面に表示される選択肢から、適切な引数を選択していきます。例えば、getRelations メ ソッドの場合、引数として RelationType が必要なので、「RelationType」を選択して、 「Next>」ボタンを押して、RelationType の一覧を表示します。その中から、収集した い RelationType を選択して、「Finish」ボタンを押すと、引数が設定できます。途中で 「<back」ボタンを押すと、一つ前の選択画面に戻ります。

「取引」タイプ

「取引」タイプは、Goodsの取引量を収集します。ここでいう「取引」とは、ある Goods を受けとったときに、特定の Goodsを送り返すというやりとりのことです。ここで収集 されるのは、あくまでも Goods であり、それをもつ Agent などは収集されません。閲覧 時には、送信された Goods の量が一列目に、返信された Goods の量が二列目に表示されます。これらは、取引ごとに行を変えて表示されます。

「取引」タイプでは、「送信 GoodsType」と「返信 GoodsType」、集計間隔を設定する 必要があります。「送信 GoodsType」や「返信 GoodsType」の設定は、「選択」ボタンを 押して、GoodsType の一覧を表示し、その中から、その取引で扱われる GoodsType を 選択して、「Finish」ボタンを押すとできます。

「取引」タイプでは、集計間隔を指定できます。1時間ステップごとに収集する場合は 「1」を、そうでない場合は任意の自然数を入力します*⁴。

「条件抽出」タイプ

「条件抽出」タイプは、元になる DataCollection から、条件に合う行のみを抽出します。 条件には、「文字列の比較」、「数値の比較」、「オブジェクトの比較」の三種類があります。

「条件抽出」タイプでは、条件と照合する列と、条件を設定する必要があります。条件と 照合する列を指定するには、「ソース」欄でその列を含む DataCollection を指定し、「対 象列」欄で列を指定します。

次に、「抽出条件」欄にある「抽出条件の設定」ボタンを押して、「抽出条件の設定」ダ イアログを開き、抽出条件を設定します。列に含まれる文字列を抽出条件とする場合は、 「文字列の比較」を選択します。同様に、列に含まれる数値を抽出条件とする場合は「数値 の比較」を、列に含まれるオブジェクトを抽出条件とする場合は「オブジェクトの比較」 を選択します。

「文字列の比較」の場合、左側の欄(最初[条件]と入力されています)に抽出基準と する文字列を入力します。そして右側の欄では、抽出基準とした文字列と完全に一致した 場合だけ抽出する(「と完全一致する」)か、文字列を含んでいる場合に抽出する(「を含 む」)かを選択します。

「数値の比較」の場合、左側の欄では等号または不等号を選択します。右側の欄には、 抽出基準となる数値を入力します。たとえば、左側の欄で「!=」を選択し、右側の欄で 「200」を入力した場合は、指定した列に「200」を含まない行が抽出されます。等号と不 等号には、次のような種類があります。

=	(抽出基準)と等しい
!=	(抽出基準)と等しくない
>	(抽出基準)より大きい
>=	(抽出基準)以上
<	(抽出基準)より小さい
<=	(抽出基準)以下

「オブジェクトの比較」の場合、指定した列のオブジェクトが null の行を抽出する(「null

^{*41}未満の数や小数を入力した場合、正常に動作しません。

である」)のか、nullではない行を抽出する(「nullでない」)のかを選択します。

抽出条件を設定したら、「抽出条件の設定」ダイアログの「了解」ボタンを押します。 「条件抽出」タイプの DataCollection は、「ソース」欄に指定した DataCollection から、 抽出条件を満たさなかった行を除いたデータになります。列の増減はありません。

「計算」タイプ

「計算」タイプは、すでにある DataCollection から、数値を含む二列を指定し、指定した二列の四則演算を行った結果を収集します。「計算」タイプには、「乗算」、「加算」、「減算」、「除算」の4種類があります。どの種類の場合も、設定の方法は変わりません。

「計算」タイプでは、計算に用いる列を2つ指定する必要があります。計算に用いる列 を指定するには、「ソース1」欄でその列を含む DataCollection を指定し、「列1」欄で列 を指定します。同様に、「ソース2」欄と「列2」欄で、二列目を指定します。計算された 結果は、次の表のように収集されます(表は加算の例)。

列1の1行目+列2の1行目
列1の1行目+列2の2行目
•
•
列2の1行目+列2の1行目
列1の2行目+列2の2行目
•
•

収集後は、計算に用いた DataCollection に結果が書き加わるのではなく、計算結果の 一列のみの DataCollection になります。

「集計」タイプ

「集計」タイプは、すでにある DataCollection から、数値を含む列を指定し、その列の 全ての行を用いた計算を行います。「集計」タイプには、「合計」、「平均」、「要素数」の3 種類があります。どの種類の場合も、設定の方法は変わりません。各種類では、次のよう な計算が行われます。

合計	列の数値の総和を計算
平均	列の数値の相加平均を計算
要素数	列に含まれる要素の数(行数)を計算

「集計」タイプでは、計算に用いる列を指定する必要があります。計算に用いる列を指 定するには、「ソース」欄で、DataCollectionを指定し、「対象列」欄で列を指定します。 「集計」タイプの DataCollection では、一列一行に計算の結果を収集します。計算に用い
た DataCollection に結果が書き加わるのではなく、計算結果が入った1つのセルのみの DataCollection になります。

6.4 収集したデータを閲覧する

収集したデータは、表形式で閲覧できます(図第 6.4): p.31。これは、閲覧する DataCollection が、どの Collector タイプを用いているかにはかかわらず、すべて表形式 で表示されます。

D取引 _ 🗆 🗙
Money
1067.0
1067.0
970.0
1261.0
4656.0
1552.0
970.0
2328.0
4850.0
9797.0
2425.0
1067.0
1940.0

図 6.4: DataCollection 収集したデータの閲覧

収集したデータを閲覧するには、DataCollection Manager の「DataCollection リス ト」でDataCollectionを選択して、「閲覧」ボタンを押します。新しいウィンドウが開か れ、収集したデータが表形式で表示されます。DataCollection の名前がウィンドウのタ イトルになっています。閲覧するDataCollection は、まとめて複数選択することが可能 ですが、「閲覧」ボタンを押したときに同じ位置に表示されるので、ウィンドウを移動し て見えるようにしてください。

列の一番上には、必ずその列の名前が表示されています。この名前は、「オブジェクト 操作」、「条件抽出」、「計算」、「集計」の各タイプを用いるときに、対象列として指定する 名前です。列の名前をドラッグすることで、列の順番を入れ替えることができます。ただ し、列の順番は一時的なものであり、保存されません。

データをダブルクリックすると、ダブルクリックしたセルがもつ属性と、属性に対応す る値が表示されます。「オブジェクト操作」タイプで「属性」を用いる場合に指定できる 属性は、ここで表示される属性です。

閲覧するデータは、各時間ステップの最後に更新されます。シミュレーションを停止中 に作成や編集をした DataCollection は、最新の情報が表示されていない場合があります。 必要に応じて、実行か一定時間実行などを行ってください(第3.1節: p.11)。

6.5 DataCollection を削除する

DataCollection を削除する場合には、DataCollection リストから、削除する DataCollection を選択して、「削除」ボタンを押します。複数の DataCollection をまとめて選択 して、削除することもできます。

6.6 データ関連の設定ファイルを扱う

6.6.1 データ関連の設定をファイルに保存する

作成した DataCollection を継続的に使用したり、他のコンピュータと共有するために、 その設定をファイルに保存することができます (Data Collection Setting 形式)。

DataCollection Manager では、DataCollection、ReportComponent(第7章: p.35)、 GraphComponent(第8章: p.41)がまとめて保存されます(ここでは、これら3つを合わせて「データ関連の設定」と呼んで、説明することにします)。

データ関連の設定を保存するには、DataCollection Manager の「保存」ボタンを押して、「Data Collector Setting Save Wizard」ダイアログ(図 6.5)で行います。

保存先とファイル名を指定した後に、どの設定を保存するかを指定します。保存する 場所を指定すると、最初の画面の「Next>」ボタンが選択可能になるので、選択して次の 画面に移ります。表示される画面は、上から「DataCollection」、「GraphComponent」、 「ReportComponent」のそれぞれのリストが表示されます。左側の「Non Save List」に ある設定は保存されず、右側の「Save List」にある設定が保存されます。移動したい設定 を選択した上で、「ADD>>」ボタンを押すと「Save List」に移動し、「<<REMOVE」ボタ ンを押すと「Non Save List」に移動します。設定は、まとめて選択することができます。

他の DataCollection を基にして作成した DataCollection を保存して使いたい場合、用 いた DataCollection も保存する必要があります。同様に、ReportComponent や Graph-Component を保存して使いたい場合も、出力する DataCollection を保存する必要があ ります。

保存するデータ関連の設定を指定したら、「Data Collector Setting Save Wizard」ダ イアログの「Finish」ボタンを押します。正常に保存処理が終わると、「Data Collection File has been Saved.」と表示されます。「了解」ボタンを押すと、データ関連の設定の保 存が完了します。

6.6.2 データ関連の設定をファイルから読み込む

DataCollection Manager では、Data Collection Setting File 形式のファイルから、 DataCollection を読み込むことができます。ファイルを読み込む場合、それに含まれて いるデータ関連の設定(DataCollection、ReportComponent、GraphComponent)がす



図 6.5: Data Collector Setting Save Wizard ダイアログ

べて読み込まれます。

データ関連の設定をファイルから読み込むには、DataCollection Manager の「読込」 ボタンを押してダイアログを表示し、ファイルを指定して「開く」ボタンを押します。読 み込まれた設定は、読み込む前からあった設定の下に並びます。



第7章

ReportComponent Manager:データ をファイルに出力する

7.1 機能と画面構成

ReportComponent Manager は、DataCollection で収集するデータを、ファイルに出力 するためのプラグインです。ReportComponent Manager では、複数の DataCollection の内容を、シミュレーションがリセットされるごとにファイルに出力することができます。 このとき、出力されるファイルは、DataCollection が収集したデータを出力した「デー タ出力ファイル」と、パラメータの初期値を出力した「世界設定出力ファイル」の2種類 です。ReportComponent Manager は、図 7.1 のような画面構成になっています。



 \boxtimes 7.1: ReportComponent Manager

各部分の機能は、以下の通りです。

① ReportComponent リスト	作成した ReportComponent が表示されます。
②「作成」ボタン	Report Component Editor を表示し、Report-
	Component を作成できます。(第 7.2 節: p.36)。
③「編集」ボタン	Report Component Editor を表示し、選択してい
	る ReportComponent を編集することができます。
	(第7.2節:p.36)。
④「動作開始」ボタン	選択している ReportComponent の動作を開始し
	て、ファイルへ出力します(第 7.3.1 節: p.38)。
⑤「動作停止」ボタン	選択している ReportComponent の動作を停止し
	ます (第 7.3.2 節: p.39)。
⑥「削除」ボタン	選択している ReportComponent を削除します(第
	7.4 節 : p.39)。

7.2 ReportComponentを作成する

7.2.1 ReportComponent の名前を設定する

ReportComponent を作成する場合は、ReportComponent Manager の「作成」ボタン を押して、Report Component Editor (図 7.2)を表示させます。

Report Component Editor
2 **
る前、Result 2 Stockの取り重
「生成ファイルの場所
基底ディレクトリ:
C:Program Files/PlatBoxSimulator
ファイル名:
☑ World名 ☑ Date ☑ Time ☑ 設定
データ出力ファイル名: ArtificialMarket_20050326_015922_ResultとStockの取引量_Data.csv
世界設定出力ファイル名: <u>ArtificialMarket_20050326_015922_ResultとStockの取引量_</u> Setting.csv
データ書き込み間隣 (Step)
1
- h=
Data Collection:
[ResultとStockの取引数]
World名 Date Time 名前

⊠ 7.2: Report Component Editor

まず、その ReportComponent がどのようなデータを出力するのかを考え、分かりやす い名前を付けましょう。Report Component Editor の「名前」欄に、任意の名前を付け ます。すでにある ReportComponent と同じ名前を付けることもできますが、特に必要 がなければ同じ名前は避けましょう。

7.2.2 ファイルの出力先と出力名を設定する

出力ファイルの場所と名前の設定は、Report Component Editor の「生成ファイルの 場所」で行います。「基底ディレクトリ」欄には、出力するディレクトリを入力します。欄 の右側にある「…」ボタンを押すと、「選択」ダイアログが表示されるので、このダイアロ グからディレクトリを選択することで、欄に場所を入力することもできます。入力が不適 切な場合には、ファイルは出力されません。

出力するファイルの名前は、「ReportComponent の名前_Data.csv」が基本となります ([ReportComponent の名前]の部分は、第7.2.1節: p.36 で設定した名前になります)。 「ファイル名」欄にあるチェックボックスにチェックを入れると、基本のファイル名にそ の文字列を加えることができます。「World 名」を有効にすると、World の名前が入りま す。「Date」を有効にすると、西暦・月・日の8桁(2005年4月1日の場合は20050401) が加わり、「Time」を有効にすると、実行時の時刻の時・分・秒の6桁(午後3時5分 20 秒の場合は150520)が加わります。これらを加えると、「World 名_Date_Time_基本 のファイル名」のように、アンダースコアで繋がれたファイル名になります。「設定」の チェックボックスは、世界設定出力ファイルの出力の有無を設定します。「設定」を有効 にすると、データ出力ファイルのファイル名の最後の部分「Data.csv」が「Setting.csv」 になったファイル名で、世界設定出力ファイルが出力されます。

7.2.3 データを出力する間隔を設定する

ReportComponent による出力の間隔を設定します。出力の間隔は、Report Component Editor の「データ書き込み間隔」欄に設定します。出力の間隔は、自然数の数字で 指定します(必ず半角文字で入力してください)。「データ書き込み間隔」で設定した時 間ごとに、ファイルが更新されます。ReportComponent の動作を途中で停止した場合で も、その時点までのデータはファイルに出力されています。

7.2.4 出力するデータを指定する

ReportComponentが出力するデータは、DataCollection によって指定します。指定した DataCollection は、「データ元」欄に表示されます。「データ元」欄の右側にある「…」 ボタンを押して、「DataCollectionを選択」ダイアログ(図7.3)を開き、DataCollection を選択します。

「DataCollection を選択」ダイアログは、左側に「DataCollection リスト」があり、右 側に「選択された DataCollection」が表示されます。左側のリストから DataCollection を選択して「ADD>>」ボタンを押すと、右の「選択された DataCollection」に移動しま

DataCollectionを選択 D	ataCollectionを選択	×
DataCollectionリスト	選択 された DataCollection	
Result と Stock の取引 Result と Stock の取引 Result と Stock の取引数 すべてのAgent すべてのAgentが所有するMoneyのQuar すべてのAgentが所有するMoney (Agen Trader AgentがらっているMoney Result と Stock の取引における Money 全 Result と Moneyの取引におけるMoney 全 すべてのAgentが所有するMoney	Result ≿ Moneyの取引数 << REMOVE	
	<u>了解</u> 取消し	

図 7.3: 「DataCollection を選択」ダイアログ

す。逆に、「<<REMOVE」ボタンを押すと、右のリストから左のリストに移動します。 DataCollectionは、まとめて選択することができます。「了解」ボタンを押すと、「選択さ れた DataCollection」にある DataCollection が「データ元」欄に登録されます。「データ 元」欄の || の中には、選択した DataCollection の名前が表示されます。

7.2.5 ReportComponent を追加する

以上のすべての設定が終わったら、Report Component Editor の「了解」ボタンを押 すと、ReportComponent リストに、その ReportComponent が追加されます。

7.3 ファイルへの出力を制御する

7.3.1 ReportComponent の動作を開始する

ReportComponent は、作成しただけでは、データをファイルに出力しません。ReportComponent の動作を開始する必要があります。ReportComponent が動作している かどうかは、ReportComponent リストに表示されています。動作している場合には、 ReportComponent の名前の後に「動作中」と表示され、動作していない場合には、「停止中」と表示されます。

停止中の ReportComponent の動作を開始するには、ReportComponent リストから ReportComponent を選択し、「動作開始」ボタンを押します。ReportComponent は、ま とめて選択することができます。ReportComponent を動作させた時点でファイルは作成 されますが、データの書き込みは、実行後、指定した時間(データを出力する間隔)が経 過したときに初めて行われます。 **ReportComponent Manager**は、ずっと表示し続けなければいけないか? 動作中に ReportComponent Manager を閉じても、ReportComponent は動作し続け ます。そのため、設定した後は、そのウィンドウを閉じても構いません。逆に、単にウィ ンドウを閉じるだけでは停止しないので、停止したい場合は必ず、ReportComponent Manager で動作を停止してください(第7.3.2節: p.39)。

シミュレーションを実行すると、動作中になっている ReportComponent が、指定した時間ごとに、データをファイルへ出力します。時間が最初に戻ると、そのたびに新しいファイルへデータが出力されます。ここで言う「時間が最初に戻る」操作には「新しく世界を開いた場合」や「制御パネルからリセットボタンを選択した場合」、「WorldInitializer で設定を適用した場合」などがあります。すでに同じ場所に同じ名前のファイルがある場合は、警告なしに上書きされるので、注意してください。

7.3.2 ReportComponent の動作を停止する

ReportComponent の動作を停止しなければ、実行して時間が進むたびにデータがファ イルへ出力されてしまうので、データの出力が必要なくなった場合には、停止するよ うにしてください。動作中の ReportComponent の動作を停止するには、ReportComponent リストから ReportComponent を選択した上で、「動作停止」ボタンを押しま す。ReportComponent の後にあった「動作中」という表示が、「停止中」に変わります。 ReportComponent は、まとめて選択することができます。

7.4 ReportComponent を削除する

ReportComponent を削除するには、ReportComponent リストから ReportComponent を選択して「削除」ボタンを押します。ReportComponent リストでは、複数の ReportComponent をまとめて選択することができます。

7.5 ReportComponent を保存する

ReportComponent の保存は、DataCollection Manager で行うことができます。これ は、ReportComponent が必ず DataCollection Manager の管理する DataCollection を 用いているためです。ReportComponent を含むデータ関連の設定の保存方法は、第 6.6 節: p.32 を参照してください。

7.6 出力されたファイルを見る

ReportComponent によってデータ出力したファイルは、指定した場所に指定した名前 で保存されています。ファイルは CSV 形式 (カンマ区切り形式) なので、テキスト・エ ディタや表計算ソフト (Microsoft Excel など)で開くことができます。

データ出力ファイルでは、一列目に時間が表示され、二列目以降は、データ元に指定した順にデータが表示されます。どの列も、一行目は列の名前になります(一列目の一行目は「Time」となります)。

世界設定出力ファイルは、一列目にパラメータの名前が表示され、二列目にパラメータの初期値が表示されます。

第8章

GraphComponent Manager:データ をグラフに出力する

8.1 機能と画面構成

GraphComponent Manager は、DataCollection が収集したデータを、グラフに出力 するためのプラグインです。GraphComponent Manager では、シミュレーションの実行 に合わせて、リアルタイムにグラフを表示します。GraphComponent は、必要なときだ け動作させることができます。

GraphComponent Manager は、図 8.1 のような画面構成になっています。それぞれの 部分の機能は、以下の通りです。

① GraphComponent リスト	作成した GraphComponent が表示されます。
② 「作成」ボタン	Graph Component Editor を表示します(第8.2節:
	p.42)
③ 「 編集 」 ボタン	Graph Component Editor を表示します(第8.2節:
	p.42),
④「動作開始」ボタン	選択している GraphComponent の動作を開始して、
	ファイルへ出力します(第 8.3.1 節: p.43)。
⑤ 「動作停止」 ボタン	選択している GraphComponent の動作を停止しま
	す (第 8.3.2 節: p.44)。
⑥ 「削除」 ボタン	選択している GraphComponent を削除します(第
	8.4 節 : p.44)。
⑦「グラフ」ボタン	選択している GraphComponent が出力したグラフ
	を表示します(第 8.6 節: p.45。



図 8.1: GraphComponent Manager の画面構成

8.2 GraphComponent を作成する

8.2.1 名前を設定する

GraphComponent を作成するには、GraphComponent Manager の「作成」ボタンを 選択するか、GraphComponent リストから GraphComponent を選択して「編集」ボタ ンを押して、Graph Component Editor (図 8.2)を表示させます。

Graph Component Editor	×
名前: New Graph	
「親データー	
Data Collection:	
0	選択
	いいえ(N)

 \boxtimes 8.2: Graph Component Editor

まず、GraphComponent がどのようなデータを出力するのかを考えて、分かりやすい 名前を付けましょう。Graph Component Editor の「名前」欄に、名前を入力します。す でにある GraphComponent と同じ名前を付けることもできますが、混乱を生じさせない ように、同じ名前は避けましょう。

8.2.2 出力するデータを指定する

GraphComponentが出力するデータは、DataCollectionによって指定します。指定した DataCollectionは、「親データ」欄に表示されます。グラフに出力されるのは、一列目の一行目(左上のセル)の値だけなので注意してください。数値以外の値が左上のセルに入力されている場合は、常に0がグラフに出力されます。

「親データ」欄の右側にある「選択」ボタンを押して、「DataCollection 選択」ダイアロ グを開き、DataCollection を選択します。「DataCollection を選択」ダイアログは、左側 に「DataCollection リスト」があり、右側に「選択された DataCollection」が表示されま す。左側のリストから DataCollection を選択して、「ADD>>」ボタンを押すと、右の「選 択された DataCollection」に移動します。DataCollection は、まとめて選択することが できます。「了解」ボタンを押すと、「選択された DataCollection」にある DataCollection は「親データ」欄に登録されます。

8.2.3 データを出力する間隔を設定する

GraphComponent による出力の間隔を設定します。出力の間隔の設定は、Graph Component Editor の「収集間隔」欄で行います。出力の間隔は、自然数の数字で指定し ます(必ず半角文字で入力してください)」「収集間隔」で設定した時間ごとに、グラフが 更新されているので、GraphComponent の動作を途中で停止した場合でも、その時点ま でのグラフは出力されています。

すべての設定が終わったら、Graph Component Editor の「了解」ボタンを選択する と、GraphComponent が GraphComponent リストに追加されます。

GraphComponent を作成する際の注意: RealClock は非対応 PlatBox で利用できる Clock には、StepClock(時間がステップで表される)と Real-Clock(時間が時刻で表される)がありますが、現在のところ、GraphComponent は RealClock には対応していません。

8.3 グラフへの出力を制御する

8.3.1 GraphComponent の動作を開始する

GraphComponent は、作成しただけでは、データをグラフに出力しません。Graph-Component の動作を開始する必要があります。GraphComponent が動作しているかど うかは、GraphComponent リストに表示されています。動作している場合には「動作 中」、動作していない場合には「停止中」と、GraphComponent の名前の後に表示され ます。

停止中の GraphComponent の動作を開始するには、リストから GraphComponent を 選択し、「動作開始」ボタンを押します。GraphComponent は、まとめて選択することが できます。GraphViewer を表示するための「グラフ」ボタン(第8.6.1節: p.45)を押し た場合にも、動作が開始します。

GraphComponent Manager は、ずっと表示し続けなければいけないか? 動作中に GraphComponent Manager を閉じても、GraphComponent は動作し続けま す。そのため、設定したら閉じてしまっても構いません。逆に、単に閉じただけでは停止 しないので、停止したい場合は必ず、GraphComponent Manager で GraphComponent の動作を停止してください(第8.3.2節: p.44)。

シミュレーションを実行すると、動作中になっている GraphComponent は、設定した 間隔ごとに、データがグラフへ出力されます。時間が最初に戻ると(リセット) そのたび に前のグラフに上書きされて出力されます。ここで、時間が最初に戻る操作には、「新しく 世界を開いた場合」、「制御パネルからリセットボタンを選択した場合」、「WorldInitializer で設定を適用した場合」などがあります。

GraphComponent の「動作中」とグラフの表示はどう違うか? GraphComponent の「動作中」とは、GraphComponent がデータを蓄積している という状態です。Graph Viewer は、その蓄積されたデータをグラフに表示します。 GraphComponent の停止中に、Graph Viewer でグラフを見ることも可能です(ただ し、データは増えません)、GraphComponent の動作中に、Graph Viewer を閉じるこ とも可能です(閉じている間もデータは蓄積されますが、見ることはできません)、

8.3.2 GraphComponent の動作を停止する

GraphComponent の動作を停止しなければ、実行して時間が進むたびにデータがグラ フへ出力されるので、データ出力の必要がなくなったら停止するようにしてください。動 作中の GraphComponent の動作を停止するには、リストから GraphComponent を選択 した上で、「動作停止」ボタンを押します。すると、GraphComponent の後ろに表示され ていた「動作中」が、「停止中」に変わります。GraphComponent は、まとめて選択する ことができます。

8.4 GraphComponent を削除する

GraphComponent を削除するには、GraphComponent リストから GraphComponent を選択して、「削除」ボタンを押します。GraphComponent リストでは、複数の Graph-

Component をまとめて選択することができます。

8.5 GraphComponent を保存する

GraphComponent の保存は、GraphComponent Manager で行うことができます。こ れは GraphComponent が必ず GraphComponent Manager の管理する DataCollection を用いているためです。GraphComponent を含むデータ関連の設定の保存方法は第 6.6 節: p.32 を参照してください。

8.6 Graph Viewer を使う

8.6.1 グラフを表示する

グラフは、Graph Viewer で見ることができます。Graph Viewer を開くには、Graph-Component Manager の GraphComponent リストからグラフを表示したい GraphComponent を選択し、「グラフ」ボタンを選択します。GraphComponent はまとめて選択で きます。Graph Viewer を開いたままシミュレーションを実行すると、時間の動きに合わ せてグラフの描画が進みますが、その分シミュレーションの実行が遅くなるので、描画の 動きを見る必要がなければ、Graph Viewer は閉じておきます。Graph Viewer を開いて いなくても、GraphComponent が動作していれば、データがグラフに出力されているの で、グラフを表示したいときに Graph Viewer を開くことができます。

Graph Viewer は、図 8.3 のような画面構成になっています。それぞれの部分の機能は、 以下の通りです。

① グラフ	グラフを描画します。X 軸ラベルと Y 軸ラベルを含み
	ます。
② 凡例	凡例を示します。「設定」ボタンを選択することでデー
	夕設定を行うことができます(第8.6.2節: p.45)
③ 簡易設定欄	グラフの簡易な設定を行います(第 8.6.3 節: p.46、第
	8.6.4 節:p.47、第 8.6.5 節:p.47)。
(4) Overview	グラフの全体像を表示します(第 8.6.6 節: p.47)。
⑤「保存」ボタン	グラフを画像ファイルに保存します(第8.6.7節: p.47)。

グラフは、X 軸(水平方向)が時間、Y 軸(垂直方向)が DataCollection が収集した データの値です。

8.6.2 Graph Viewer のデータ設定を行う

初期設定では、グラフはデータごとに異なる色の線で描画されます。また、データが複数ある場合には、X 軸、Y 軸はそれぞれ最初のデータの軸に連動するように統合されています。線の色と、軸の統合に関する設定は、「凡例」のところにある「設定」ボタンを押



図 8.3: Graph Viewer の画面構成

して、「データ設定」ダイアログを開きます。

「データ設定」ダイアログには、「名前」、「線の色」、「X軸の統合」、「Y軸の統合」の欄 があります。「名前」は、DataCollectionの名前です(変更はできません)」「線の色」は、 描画する線の色です。「線の色」のボタンを押すと、「色の選択」ダイアログが表示される ので、任意の色を指定します。

「X軸の統合」、「Y軸の統合」では、軸の統合を設定できます。どの軸と統合するかを チェックボックスで選択します。初期設定では、一つ上の軸に統合する設定になってい ます。

2つ目以降のチェックをしても、統合する軸を選択しなかった場合は軸の統合は無効に なります。また、すべての軸の統合を有効にすると(少なくとも一つの軸は、どの軸に統 合してよいか分からなくなるため)、一番上の設定が赤くなります。この警告を無視する と、赤くなったデータの設定は無効になります。

設定が終わったら、「了解」ボタンを押して、設定を有効にします。

8.6.3 グラフの軸の倍率を調節する

グラフの倍率によっては、見たいデータが見にくい場合があります。Graph Viewer で は、各データの軸ごとにグラフの倍率を変更することができます。

初期設定では、Graph Viewer の簡易設定欄にある「X 軸自動倍率調節」と「Y 軸時自動倍率調節」が有効になっています。自動倍率調節は、すべての値がグラフ領域に収まるように、軸の倍率を自動的に調節する機能です。

自動倍率調節を無効にしている場合は、軸ラベルにある「+」「・」「・」の各ボタン で軸の倍率を調節することができます。「+」を押すと、倍率が2倍になり、「-」を選択 押すと、倍率が2分の1になります。「・」を押すと「Axis Setting」ダイアログが表示さ れ、軸の目盛りと倍率を調節できます。「Axis Setting」ダイアログの「目盛り」と「倍 率」はともに「自動」か「マニュアル設定」かを選択できます。「マニュアル設定」の場 合、欄に任意の自然数を入力して、「適用」ボタンを押すと、「Axis Setting」ダイアログ を表示したままで「マニュアル設定」を反映できます。「了解」ボタンを押すと、設定が 有効になり、「Axis Setting」ダイアログが閉じます。

8.6.4 グラフのX軸の動きを自動的に追跡する

X 軸の倍率を自動調節にしていない場合、時間がしばらく経過すると、データが Graph Viewer のグラフ領域の外に出てしまう場合があります。Graph Viewer では、X 軸の 最新の値がグラフ領域に収まるように、X 軸の動きを追跡することができます。Graph Viewer の簡易設定欄で、「X 軸自動追跡」を有効にすると、X 軸の動きを追跡します。

8.6.5 グラフの罫線を表示する

Graph Viewer では、グラフに罫線を引くかどうかを、軸ごとに設定できます。Graph Viewer の簡易設定欄で、「X軸の罫線表示」や「Y軸の罫線表示」を有効にすると、軸ラベルの目盛りに合わせて罫線が表示されます。

8.6.6 グラフの全体像を見る

軸の倍率を自動調節にしていない場合、グラフの全体像が見えない場合があります。 Graph Viewer の Overview 欄には、グラフの全体像が常に表示されています。Overview 欄で赤い枠に囲まれているのが、グラフに表示されている領域です。Overview 欄の中で クリックまたはドラッグすると、マウス・ポインタの位置を中心に表示位置が移動します。

8.6.7 グラフを画像ファイルに出力する

Graph Viewer では、グラフを JPEG 形式の画像ファイル(拡張子.jpg)として保存で きます。シミュレーションの時間を最初に戻すと、そのたびにグラフは初期化されてしま うので、必要であれば保存してください。Graph Viewer の Overview 欄の下にある「保 存」ボタンを押すと、「画像出力の設定」ダイアログが表示されます。そこで画像のサイ ズを設定し、保存先を指定することで、グラフを画像ファイルとして保存できます。

「画像出力の設定」ダイアログでは、画像サイズを「幅」と「高さ」のピクセル数で指定 します。初期設定で入力されている値は現在の倍率で、描画しているすべてのデータを収 めることができるサイズです。ただしグラフの表示領域が、そのサイズを超えている場合 はグラフのサイズが初期設定に入力されています。サイズは変更できますが、初期設定値 を超えるピクセル数を指定することはできません。

「画像出力の設定」ダイアログで「OK」ボタンを押すと、「保存」ダイアログが表示されます。「保存」ダイアログで、「ファイル名」と「保存する場所」を指定して、「保存」ボタンを押すと、グラフが画像ファイル (JPEG 形式) として保存されます。画像ファイルには、目盛りや罫線は保存されますが、倍率調節のボタンや軸ラベルは保存されません。

第9章

Log Monitor: ログをみる

9.1 機能と画面構成

Log Monitor は、シミュレーションのログを出力・閲覧するためのプラグインです。モ デル作成時に、出力したい情報とその重要度(出力レベル)をあらかじめ指定しておくこ とによって、実行時に指定した出力レベルに応じて、ログの内容を出力することができ ます。

Log Monitor は、図 9.1 のような画面構成になっています。各部分の機能は、以下の通 りです。

メニュー・表示設定	ログの表示/非表示を、出力レベル別やパッケージ/ク
	ラス別で設定できます。出力レベルは、5 段階あります
	(FATAL, ERROR, WARN, INFO, DEBUG).
表示選択パネル	ログの表示/非表示を、パッケージやクラス単位で設定で
	きます。
ログー覧表示パネル	ログの一覧が表示されます。表示数の上限は、初期値では
	5000 ですが、メニューの [Configure]-[Set Max Number
	of Records] から変更できます。ログ項目の表示/非表示
	は、メニューの [View] から設定できます。
詳細ログ表示パネル	ログー覧表示パネルで選択されたログの内容が、詳細表
	示されます。

9.2 出力されるログの項目

詳細ログ表示パネルに、詳細表示されるログの各項目は、以下の通りです。

ories	Date	Thread	Message #	Level	NDC	Category	Message	Location	Thrown		
5 org	Thu Mar	Thread-1	142	INFO		foreignexchangemarket.mode.	Create 投機家 Agent	foreignexcha.			
	Thu Mar	Thread-1	143	INFO		foreignexchangemarket.mode.	Create 投税家 Agent	foreignexcha			
	Thu Mar	Thread-1	144	INFO		foreigneix:hangemarket.mode.	Create 投程家 Agent	foreignexcha.			
	Thu Mar	Thread-1	145	INFO		foreignexchangemarket.mode.	Create 投稿家 Agent	foreignexcha.			
	Thu Mar	Thread-1	146	INFO	-	foreignexchangemarket mode.	Create 投稿家 Agent	foreignexcha			
	Thu Mar	Thread-1	147	INFO		foreignexchangemarket mode.	Create 194818 Agent	foreignexcha			ーロクー寛衣ホハー
	Thu Mar	Thread-1	140	INFO		foreignexchangemarket.mode.	Create 投稿家 Agent	foreignexcha.			
	Thu Mar	Thread-1	149	INFO		foreignexchangemarket.mode.	Add Relations	foreignexcha			
	Thu Mar	Thread-1	150	INFO		foreignexchangemarket mode.	add NewRelationGroup.	foreignexcha			
	Thu Mar	Thread-1	151	INFO		foreignexchangemarket.mode.	add NewRelationGroup.	foreignexcha.			
	Thu Mar	AWT-Eve.	152	INFO		org platbox simulator present.	PresentationCompone.	org.platbox.s.		1	
	Date: Thread: Message I: Level: NDC: Category: Message: Location: Thread:	Thu Mar main : 1 INFO org.plat Log Init org.plat	24 18:28:22 J box.simulator ialized by g: box.simulator	st 2005 contain il contain	(11116 wer.Plo wer.Pla	60102903) (BoxSimulator (BoxSimulator, initializekog(Pi	atBoxSimulator.java;184	>			- ログ詳細表示パ

⊠ 9.1: Log Monitor

Date	ログが出力された日時
Thread	ログ出力元のスレッド(特に参照する必要はありません)
Message $\#$	ログの番号
Level	ログの出力レベル
NDC	ネスト化診断コンテキスト (特に使用しません)
Category	ログ出力元の Java クラス
Message	ログの内容 (モデル作成者の設定した出力内容)
Location	ログ出力元のメソッド
Thrown	発生した例外処理 (Exception や Error の内容)

第10章

Profiler:メモリの状態を見る

10.1 機能と画面構成

10.1.1 Profiler でできること

Profiler は、実行中のモデルに存在するオブジェクトの数や、メモリの状態を監視する ためのプラグインです。Profiler は、以下のような問題の原因を突き止めるために用い ます。

- ステップが進むにつれて、1 ステップあたりの実行速度が低下する。
- OutOfMemoryError^{*1}によって、シミュレーションが強制的に停止する。

これらの問題に対して、以下のような原因が考えられます。Profilerは、これらの原因の特定に役立ちます。

- 削除すべき Agent/Relation/Channel などのオブジェクトが、きちんと削除されていない。
- Behavior/Information などのオブジェクト内部で、大量のデータが蓄積されている。
- 外部へのデータ出力 (コンソール出力など)の処理が、メモリを大幅に利用している。

Profiler は、図 10.1 のような画面構成になっています。各部分の機能は、次節以降で説明します。

10.1.2 操作パネル

(1) Update Interval 設定フィールド, START/STOP ボタン

「Memory History 表示パネル」と「Instance History 表示パネル」の更新頻度を設定 します。各設定フィールドに値を入力した後、[START] ボタンを押すと、設定が反映さ

^{*&}lt;sup>1</sup> メモリ不足のために Java 仮想マシンがオブジェクトを割り当てることができず、ガベージコレクタに よっても使用可能なメモリをこれ以上確保できない場合にスローされる Exception です。

	Profiler					
	Profiler state:RUNNING Me	mory History state:STOP	PPED Instance History state	STOPPED		
操作パネル	Update Interval(ms) 500	Interval(ms) 30000	Interval(step) 10	GC S	napShot	
	STOP	START	START			
	Class		Alive	Allocate	d	
	org.platbox.queuetheory.model.Airport/Vorld	1		1		
	org.platbox.queuetheory.model.behavior.Depart	1		1		
	org.platbox.queuetheory.model.behavior.DropP	a 1		1		
1	org.platbox.queuetheory.model.behavior.Inform	a 1		1		
Instance History	org.platbox.queuetheory.model.behavior.Manag	e6		6		
表示パネル	org.platbox.queuetheory.model.behavior.moveA			72		
10,10,1-10	org platbox simulator model timfw Agent	87		87		
	org platbox simulator model fmfw Channel	361		1023		
	org.platbox.simulator.model.fmfw.Goods	491		1495		
	org.platbox.simulator.model.fmfw.Relation	369		485		
· · · · · · · · · · · · · · · · · · ·		· ·				
Memory History		dax Memory	Total Memory	Free Memory		
表示パネル	66,650,11	2 4,1	714,496 5	09,600		

図 10.1: Profiler ウィンドウ

$nst s^{*2}$

Profiler Update Interval (ms)	全ての表示パネルの更新頻度を、ミリ秒単
	位で設定します。
Memory History Interval (ms)	Memory History 表示パネルの更新頻度
	を、ミリ秒単位で設定します。
Instance History Interval (step)	Instance History 表示パネルの更新頻度
	を、ステップ単位で設定します。

(2) GC ボタン

どのオブジェクトからも参照されていないオブジェクトを消去して、メモリ領域を増や します (この機能はガベージコレクションと呼ばれます)。Java は自動的にガベージコレ クションを行うため、モデル作成者がこの機能を意識する必要はありませんが、現在のオ プジェクトの数を正確に知りたい場合は、このボタンを押してください。

(3) SnapShot ボタン

Profiler の表示を更新します。シミュレーション停止中に、最新のオブジェクト数や残 りメモリ領域の値を知りたい場合は、このボタンを押してください。

10.1.3 Instance History 表示パネル

(1) Class

読み込まれているモデルで使用されているクラスの一覧が表示されます。

^{*2} 各表示パネルの更新頻度が複数のフィールドより重複して設定された場合は、更新頻度の短い設定が優先 されます。

(2) Alive

現時点のオブジェクトの数が表示されます。ステップが進むにつれてこの値が増加して いる場合は、メモリ不足の原因としてオブジェクト数の増加(モデル作成者が意図した通 りにオブジェクトが削除されていない)が考えられます。

(3) Allocated

モデルが実行されてから現時点までに生成されたオブジェクトの合計数が表示されま す。合計数という性質上、この値が減少することはありません。ステップが進むにつれ て、この値が極端に増加している場合は、そのオブジェクトを生成する処理が冗長である という可能性が考えられます。

10.1.4 Memory History 表示パネル

(1) Max Memory

Java 仮想マシンが使用を試みる最大メモリ容量が表示されます。

(2) Total Memory

Java 仮想マシンのメモリの総容量が表示されます。コンピュータの状況によって、この数値は時間とともに変化する場合があります。

(3) Free Memory

Java 仮想マシン内の空きメモリの量が表示されます。この数値は、メモリが不足して いるかどうかを判断するための指標となります。

10.2 Profiler による観察結果の解釈

Profiler による観察結果をどのように解釈するのかについて、いくつかの例を挙げて説明します。

10.2.1 オブジェクト数が増加して、空きメモリ量が減少している場合

実行速度低下・メモリ不足の問題があるとき、ステップが進むにつれて、 Agent/Relation/Channel などのオブジェクト数 (Alive) が増加し、空きメモリの 量 (Free Memory) が減少している場合には、「オブジェクト数の増加」が原因であると考 えられます。

このとき、どのクラスのオブジェクトが増加しているのかによって、対応策が異なりま す。以下では、増加しているオブジェクトのクラス別に、対応策を示します。 (1) Agent のオブジェクト 数が増加

モデル作成者の意図した通りに、Agent が追加/削除されていない可能性があります。 Agent を追加/削除する処理を確認して、正しく追加/削除が行われているか確認してください。

また、削除される Agent への Relation が、削除されていない可能性があります。Agent のがもつ Relation を確認して、削除される Agent への Relation を削除するようにして ください。

(2) Relation のオブジェクト 数が増加

モデル作成者の意図した通りに、Relationが追加/削除されていない可能性があります。 Relationを追加/削除する処理を確認して、正しく追加/削除が行われているかを確認し てください。

(3) Channel のオブジェクト 数が増加

Agent が Goods/Information を送信するときに、Channel を閉鎖しないように設定さ れている可能性があります。Agent が Goods/Information を送信する処理を確認してく ださい。アクションパーツの引数として、keepChannel を true に設定している場合、開 設された Channel は明示的に閉鎖されるまで、存在し続けます。特に、Channel を保持 しておく必要がない場合は、keepChannel を false に設定するか、自動的に閉鎖されるア クションパーツを利用してください。

(4) Information のオブジェクト 数が増加

Information のオブジェクトが、追加され続けている可能性があります。不必要なオブ ジェクトは削除して、共有できるオブジェクトは統合してください。

10.2.2 オブジェクト数は増加していないが、空きメモリ量が減少してい る場合

実行速度低下・メモリ不足の問題があるとき、ステップが進むにつれて、 Agent/Relation/Channel/Information などのオブジェクト数 (Alive) は増加して いないが、空きメモリの量 (Free Memory) が減少している場合は、「PlatBox 基礎モデル に基づかないオブジェクトの増加」や「コンソールなどへの外部出力処理」が原因である と考えられます。Profiler 上では、これ以上原因を絞り込むことはできません。これ以上 の探索は、以下のように行います。 (1) PlatBox 基礎モデル以外のオブジェクト 数の増加

基礎モデル以外のクラスのオブジェクト数が増加している可能性があります。まず、 Information/Behaviorの設計を確認して、ステップ数が進むにつれて、オブジェクト数 やデータ量が増加する設計になっているかどうかを確認してください。例えば、人工市場 モデルにおいては、毎ステップごとの株価を時系列データとして Information に保持する ことなどが、これにあたる。

もしそのように設計されている場合は、それらのオブジェクト数やデータ量が増加しな いように、モデルを修正してください。例えば、古い時系列データは使用しないのであれ ば、明示的に削除してください。また、複数の Agent で共有できるデータ (例えば株価な ど)であれば、1 つの Agent に集約して記憶させておき、他の Agent が必要に応じて問い 合わせる、というモデル化を行ってください^{*3}。

(2) コンソールなどへの外部出力処理

コンソールや外部ファイルなどへデータを出力する処理が、行われている可能性もあり ます。もしそのような処理が行われている場合は、その部分を削除し、再度シミュレー ションを実行してください。モデルの各部分でコンソール出力(標準)を頻繁に行ってい る場合や、独自に設計された外部ファイルへの出力処理を行っている場合には、メモリ不 足が生じることがあります。

^{*&}lt;sup>3</sup>ただし、このモデル化では、Agent 同士のコミュニケーションの回数が増加するため、メモリ不足の問題 が解消されても、コミュニケーション回数の増加によって実行速度が低下する可能性があります。

第11章

Communication Viewer: Agent のや りとりを見る

11.1 機能と画面構成

Communication Viewer では、シミュレーションを実行する際に行われる Agent 間の Goods や Information などのやりとりを、アニメーション^{*1}によって表示することがで きます。Communication Viewer は、図 11.1 のような画面構成になっています。それぞ れの部分の機能は、以下の通りです。



⊠ 11.1: Communication Viewer

^{*&}lt;sup>1</sup> 複数の静止画像(コマ)によって動きを表現することを指します(ここでいうアニメーションとは、具体 的には「Goodsの送信」、「Channelの接続/切断」、「Relationの追加/削除」、「Behaviorの状態遷移」、 「TimeEventの受信」の5項目を指します)。

表示領域 Agent、Behavior、Relation、Channel を表示します。
設定パネル 表示/非表示の設定欄と、アニメーションの表示/非表示の

パネル 表示/非表示の設定欄と、アニメーションの表示/非表示の設 定欄が含まれます。表示領域に表示する部品と、アニメーションの設定を行います(第11.2節: p.58、第11.3節: p.58)。

表示領域には、図 11.2 に示す部品が配置されます。



図 11.2: Communication Viewer 表示パネルの部品

Agent	Agent を示す画像と「AgentType のラベル」で表現されます。
	ドラッグして移動できます。
Behavior	Agent から出る四角形の吹き出し内で、色つきの四角形に表示
	されます。Behavior 名と「状態」で表現されます。青に点滅し
	ている場合は TimeEvent を受信していることを示し、赤に点
	滅している場合は状態遷移が起こっていることを示します。
Relation	「RelationType のラベル」と Relation を示す破線と矢印で表
	現されます。
Channel	太い実線で表現されます。

11.2 表示を設定する

「設定パネル」の上側にある表示/非表示の設定欄では、「AgentType のラベル」、 「Relation」、「RelationType のラベル」、「Behavior」、「状態」の5つの表示/非表示を 設定することができます。チェックボックスにチェックを入れたものが表示され、チェッ クをはずしたものは表示されません。

11.3 アニメーションの表示を設定する

「設定パネル」の下側にあるアニメーションの表示/非表示の設定欄では、アニメーションの表示/非表示と、速度を設定することができます。この設定には、全てのアニメーショ

ンの一括設定と、個別のアニメーションの設定があります。全てのアニメーションの設定 項目としては、「スキップ」と「全てのアニメーションを表示」があり、個別のアニメー ションの設定項目としては、「表示/非表示」、「ループ」、「滑らかさ」、「点滅回数」、「間隔」 があります。

11.3.1 全てのアニメーション

「全てのアニメーションを表示」を設定することができます。

スキップ	現在行われているアニメーションの続きを省
	略し、次のアニメーションを表示します。
全てのアニメーションを表示	個別のアニメーションの各項目の設定を有
	効にすることができます。Communication
	Viewer を開いた時には、チェックが入ってい
	ます。チェックをはずすと、各項目における
	設定が無効になり、アニメーションは表示さ
	れません。

11.3.2 個別のアニメーション

「Goods の送信」、「Channel の接続/切断」、「Relation の追加/削除」、「Behavior の状 態遷移」、「TimeEvent の受信」の各項目の設定を変更することができます。

表示/非表示	(全ての項目)	各項目の表示/非表示を設定することができま す。チェックボックスにチェックを入れたも のは表示領域に表示され、チェックをはずし たまのは表示されません
ループ	(全ての項目)	任意の項目のアニメーションを繰り返すよう に設定することができます。ループのチェッ クボックスにチェックを入れると、そのチェッ クを入れた項目のアニメーションを繰り返し ます。例えば、「Goods の送信」の項目にあ るループのチェックボックスにチェックを入 れた場合、Behavior などの「Goods の送信」 以外のアニメーションは影響を受けませんが、 「Goods の送信」のアニメーションを繰り返 します。
滑らかさ	(「Goods の送 信」、「Channel の接続/切断」、 「Relation の追 加/削除」)	アニメーションの滑らかさ ^{*2} を設定すること ができます。スライダーを動かして設定値 を変更します。設定値を小さくするほどアニ メーションの動きは粗くなり、速くなります。
点滅回数	(「Behavior の状態遷移」、 「TimeEvent の受信」)	「Behavior の状態遷移」や「TimeEvent の 受信」が行われた際の Behavior の点滅回数 を設定することができます。スライダーを動 かして設定値を変更します。1回の点滅時間 は変わらないので、設定値を小さくするほど アニメーションは速くなります。
間隔	(全ての項目)	アニメーションのコマの間隔を設定すること ができます。スライダーを動かして設定値を 変更します。設定値を小さくするほど、アニ メーションは速くなります。

第12章

Relation Viewer: Agent 間の Relation を見る

12.1 機能と画面構成

Relation Viewer は、Agent 間の Relation の結ばれ方を見るためのプラグインです。 Relation Viewer では、「円形配置」、「レイヤー配置」、「ランダム配置」、「線形グラフ」、 「両対数グラフ」の5種類の方式で、Relation の結ばれ方を見ることができます。ビュー アに表示する AgentType と RelationType を指定することができます。

表示領域に表示する AgentType が一覧表示され
ます。
表示領域に表示する RelationType が一覧表示さ
れます。
Agent と Relation が表示されます。表示には「円
形配置」、「レイヤー配置」、「ランダム配置」、「線
形グラフ "「両対数グラフ」の 5 種類があります。

12.2 表示項目を設定する

12.2.1 表示する Agent を設定する

表示領域に表示する AgentType を設定することで、余計な Agent を描画しないようにできます。描画する AgentType は「AgentType リスト」に表示されます。初期設定では描画できるすべての AgentType が設定されています。描画する AgentType は、「AgentType リスト」右下の「Arrange」ボタンを選択すると表示される「Type Chooser」 ダイアログで設定します。

「Type Chooser」ダイアログ左側の「Hide Types」にある AgentType は描画されず、 「Show Types」ににある AgentType は描画されます。移動したい AgentType を選択し た上で、「ADD>>」ボタンを選択すると「Show Types」に移動し、「<<REMOVE」ボタ



☑ 12.1: Relation Viewer

ンを選択すると「Hide Types」に移動します。AgentType はまとめて選択することがで きます。「Type Chooser」ダイアログの「了解」ボタンを選択すると「AgentType リス ト」が更新されます。「取消し」ボタンを選択すると「AgentType リスト」は更新されま せん。

World を読み込んだとき、AgentType リストには、その時点で World に存在してい る Agent がいる AgentType しか登録されていません。その時点で存在しない Agent お AgentType を描画するには、「Hide Types」から「Show Types」に移動してくださ い。Relation Viewer では、描画される Relation を一つももたない Agent であっても、 「AgentType リスト」に登録された AgentType であれば描画されます。

12.2.2 表示する Relation を設定する

表示領域に表示する RelationType を設定することで、余計な Relation を描画しな いようにできます。描画する RelationType は「RelationType リスト」に表示されま す。初期設定では描画できるすべての RelationType が設定されています。描画する RelationType は、「RelationType リスト」右下の「Arrange」ボタンを選択すると表示 される「Type Chooser」ダイアログで設定します。

「Type Chooser」ダイアログ左側の「Hide Types」にある RelationType は描画されず、 「Show Types」ににある RelationType は描画されます。移動したい RelationType を選 択した上で、「ADD>>」ボタンを選択すると「Show Types」に移動し、「<<REMOVE」ボ タンを選択すると「Hide Types」に移動します。RelationType はまとめて選択すること ができます。「Type Chooser」ダイアログの「了解」ボタンを選択すると「RelationType リスト」が更新されます。「取消し」ボタンを選択すると「RelationType リスト」は更新 されません。

Relation Viewer では、結び先と結び元の両方の Agent が描画されない Relation は、 「RelationType リスト」に登録されていても描画されません。

12.3 **いろいろな方法で** Relation を見る

12.3.1 Agent を円形に並べる

Agent 間の Relation がどのような構造なのかを見るには、Agent を円形に配置し、 Agent 間の Relation を線で描画する方式が適しています図 12.2。この方式の場合表示領 域上での Relation どうしの重なりが少なく、Relation の集中している Agent が見つけや すいといった長所があります。



図 12.2: Relation Viewer 円形配置

Relation Viewer の表示領域で Agent を円形に配置するには「円形配置」タブを選択 します。表示領域には Agent が円周上に等間隔に黒い丸で描画され、Relation は Agent 間の線として描画されます。Agent には AgentType 名が近くに記述されます。Relation の線には Relation の方向を示す矢印が描画されます。また、シミュレーションの停止中 に Agent をドラッグすると Agent を表示領域上の任意の場所に配置できます。シミュ レーションの時間を動かしても、ドラッグで移動した Agent は円に戻らず配置した場所に描画されます。

Agent は、「AgentType リスト」の上に表示されている AgentType の Agent から順 に、円の上(時計の0時の位置)から時計回りに配置されます。シミュレーションの実行 中に Agent が追加された場合、その AgentType の最後尾に、追加された Agent が配置 されます。

12.3.2 AgentType ごとに一列に並べる

AgentType 間の Relation がどのように結ばれているかを見るには、Agent を Agent-Type ごとに層を分けて描画する方式が適しています図 12.3。この方式の場合、Agent が AgentType ごとに分かれるので、AgentType による Relation の結ばれ方の違いや層構 造が把握しやすいといった長所があります。





図 12.3: Relation Viewer レイヤー配置

Relation Viewer で AgentType ごとに一列に並べて配置するには「レイヤー配置」タ プを選択します。表示領域には Agent が AgentType ごとに等間隔に黒い丸で描画され、 Relation は Agent 間の線として描画されます。Agent には AgentType 名が近くに記述 されます。Relation の線には Relation の方向を示す矢印が描画されます。また、シミュ レーションの停止中に Agent をドラッグすると Agent を表示領域上の任意の場所に配置 できます。シミュレーションの時間を動かしても、ドラッグで移動した Agent は列に戻 らず配置した場所に描画されます。ただし、Agent 以外の場所をクリックしたり、シミュ レーションの時間を動かすとすべての Agent の配置は初期化されます。

Agent は、「AgentType リスト」の上にある AgentType ごとに上から一列に並べて配置されます。シミュレーションの実行中に Agent が追加された場合、その AgentType の 最後尾に追加された Agent が配置されます。Agent が増えて表示領域の横幅が足りなく なると、自動的に Agent 間の幅を縮めて、すべての Agent が描画領域に収まります。

12.3.3 Agent を自由に配置する

Agent を任意に配置して、Relation の追加・削除や組み替えの様子を見るには、Agent を自由に配置して描画する方式が適しています図 12.4。この方式の場合、見たいことに応じて自分で好きなように Agent を配置できるので、円形配置やレイヤー配置では分かりづらい構造を捉えることができるといった長所があります。



図 12.4: Relation Viewer ランダム配置

Relation Viewer の表示領域で Agent を自由に配置するには「ランダム配置」タブを選 択します。表示領域には Agent がランダムに黒い丸で描画され、Relation は Agent 間の 線で描画されます。また、シミュレーションが実行中か停止中かにかかわらず、Agent を ドラッグすると Agent を表示領域上の任意の場所に配置できます。この配置は表示領域 上の Agent 以外の場所をクリックしたり、シミュレーションを実行した場合も維持され ます。

Agent は Relation Viewer の表示領域上にランダムに配置されます。一度ドラッグで 移動した Agent はその位置に固定されます。なお、Relation Viewer の表示領域の大きさ を変更しても、Agent の位置は変わりません。

12.3.4 Agent を Relation 数の順に並べる

それぞれの Agent がどの程度の Relation を持っているのか、Agent の Relation 数は どういった傾向があるのかを見るには、持っている Relation の数で Agent に順位を付け て、順にグラフ上に描画する方式が適しています図 12.5。この方式の場合、Agent ごと の Relation 数の格差が見られるほか、べき乗分布*1の有無を見られるといった長所があ ります。



図 12.5: Relation Viewer グラフ

Relation Viewer の表示領域で Agent を Relation 数の順にグラフ上に配置するには「線形グラフ」タブまたは「両対数グラフ」タブを選択します。表示領域には X 軸(水平 方向)に順位(Ranking)を、Y 軸(垂直方向)に Relation 数(Link 数)を取ったグラ フが表示され、Agent は黒い丸でグラフ上に描画されます。Agent 間の Relation は描画 されません。また、Agent の配置は自動的に行われ、ドラッグなどで移動することはでき ません。

「線形グラフ」は X 軸、Y 軸ともに等間隔に目盛りが取られていて、「両対数グラフ」は X 軸、Y 軸ともに一目盛り進むごとに値が 10 倍になるように取られています。両対数グ ラフはべき乗分布の有無が見られます。

^{*1} 両対数グラフ上で直線で表されるような分布


PlatBox Project では、シミュレーション・プラットフォーム「PlatBox Simulator」、 およびシミュレーション作成支援ツール「Component Builder」を開発・提供していま す。 PlatBox Simulator は、シミュレーションを「動かす」ことによって、社会を理解す るための道具です。これに加えて、私たちは、シミュレーションを「つくる」過程も社会 を理解する助けになると考えています。「Component Builder」は、そのような理解を助 けるモデル作成支援ツールです。この「動かすことでわかる」と「つくることでわかる」 という二つのアプローチによって複雑な社会を理解する そのための「新しい思考 の道具」をつくることが、私たちの目指すゴールです。

PlatBox Simulator は、Boxed Economy Project によって開発されてきた「Boxed Economy Simulation Platform」(BESP)の発展版です。Boxed Economy Project は、 1999 年から、社会・経済のモデル化を支援する道具立てについて考えてきました。その 後、私たちが提案するモデル化の枠組みは、経済分野に限定されたものではないと考え、 2005 年にプロジェクト名を「PlatBox Project」に、ソフトウェアの名前を「PlatBox Simulator」に変更しました。

【Boxed Economy Project 時代の主なメンバー】

- 青山希(2002年~現在)
- 浅加 浩太郎 (2001 年)
- 井庭 崇(1999 年~現在) プロジェクトリーダー
- 海保研(2000年~2002年)
- 上橋 賢一(2000 年~2002 年)
- •北野里美(2001年~2002年)
- 高部 陽平 (1999 年~2001 年)
- 武田 林太郎 (2003 年~2005 年)
- 田中 潤一郎 (2000 年~2002 年)
- 中鉢 欣秀(2000 年~2001 年)
- 津屋 隆之介(2000年~2005年)
- 広兼 賢治(1999年~2001年)
- 松澤 芳昭 (2000 年~2004 年)
- 森久保 晴美(2001 年~2002 年)
- 山田悠(2001年~2005年)



「PlatBox Simulator マニュアル」

執筆・編集者

PlatBox Project (編著)

青山 希

笠井 賢紀

津屋 隆之介

宮澤 かおり

伊藤 諭志

井庭 崇

Designers' Guide to Social Simulation, No.1 PlatBox Simulator $\forall \exists \exists \mathcal{T} \mathcal{W}$

2005 年 4 月 1 日 初版発行 2005 年 8 月 27 日 第 2 版発行

編著: PlatBox Project

〒252-8520 神奈川県藤沢市遠藤 5322 慶應義塾大学 井庭崇研究室 E-Mail: platbox@sfc.keio.ac.jp, もしくは iba@sfc.keio.ac.jp Web: http://www.platbox.org/

